# Scaling BPM Adoption

## From Project to Program with IBM Business Process Manager

- Embrace process awareness as your roadmap for change
- Design robust business processes that scale with your needs
- Succeed with prescriptive methods and guidelines

Lisa Dyer
Flournoy Henry
Ines Lehmann
Guy Lipof
Fahad Osmani
Dennis Parrott
Wim Peeters
Jonas Zahn

Redbooks

ibm.com/redbooks

International Technical Support Organization

# Scaling BPM Adoption: From Project to Program with IBM Business Process Manager

March 2012

SG24-7973-01

**Note:** Before using this information and the product it supports, read the information in “Notices” on page vii.

**Second Edition (March 2012)**

This edition applies to Version 7, Release 5, of IBM Business Process Manager.

# Contents

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
*IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.*

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:** INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

Blueworks Live™
DataPower®
Global Business Services®
IBM®
Rational Team Concert™
Rational®
Redbooks®
Redbooks (logo) ®
WebSphere®

The following terms are trademarks of other companies:

Microsoft, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Snapshot, and the NetApp logo are trademarks or registered trademarks of NetApp, Inc. in the U.S. and other countries.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries. Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

# Preface

Your first Business Process Management (BPM) project is a crucial first step on your BPM journey. It is important to begin this journey with a philosophy of change that allows you to avoid common pitfalls that lead to failed BPM projects, and ultimately, poor BPM adoption. This IBM® Redbooks® publication describes the methodology and best practices that lead to a successful project and how to use that success to scale to enterprise-wide BPM adoption. This updated edition contains a new chapter (Chapter 4, “Planning a BPM project” on page 97) on planning a BPM project.

The intended audience for this book includes all people who participate in the discovery, planning, delivery, deployment, and continuous improvement activities for a business process. These roles include process owners, process participants, subject matter experts (SMEs) from the operational business, and technologists responsible for delivery, including BPM analysts, BPM solution architects, BPM administrators, and BPM developers.

## The team who wrote this book

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

**Lisa Dyer** is the community manager for IBM Business Process and Decision Management. She has seven years of experience with BPM and more than 10 years of experience in software development. Her areas of expertise include providing knowledge and other resources that help solution teams implement their BPM projects. Lisa holds a degree in Design from the College of Arts in Ikaalinen, Finland. She has written extensively about business process implementation.

**Flournoy Henry** is a Vice President at BP3, a Business Process Company and IBM Business Partner. He has over 12 years of experience in Process and Workflow Management. He holds a bachelors of science in business administration with a concentration in Management Information Systems from Auburn University. His passion for BPM as a discipline, and not just a software tool, helps bridge the gap between business value and IT deliverables. He's been a guest speaker at several Gartner and BPM User conferences and written on various BPM topics.

**Ines Lehmann** is a Business Process Management (BPM) Consultant at the IBM Development Laboratory in Boeblingen, Germany. She is part of the IBM Software Services for WebSphere® team and has been supporting customers in business process integration projects for more than five years. Ines has more than 7 years of customer experience in software development projects. She holds a degree in Business Information Systems from Reutlingen University and a Master's degree from Bamberg University. Her areas of expertise include service-oriented architecture (SOA) solutions, Business Process Management, IBM Business Process Manager, WebSphere Process Server, WebSphere Integration Developer, WebSphere Business Monitor, and Business Space. Ines is an author of *Human-Centric Business Process Management with WebSphere Process Server V6*, SG24-7477, and has written extensively on BPM implementation topics.

**Guy Lipof** leads the BPM Project Management Office. He has over five years of experience delivering BPM projects and building BPM programs and over twenty years of enterprise software development experience. Guy is an expert practitioner in agile software development management and graduated from the University of Texas at Austin. Before leading the IBM Business Process Manager PMO, he lead BPM Program Management for Lombardi EMEA. His areas of expertise include business process analysis, optimization, agile project management, and BPM program development. Guy specializes in large complexity projects and guiding customers from project to program.

**Fahad Osmani** is a BPM Solution Architect for IBM North America. He has 8 years of experience working with BPM methods and technology and has worked in the software industry over 10 years. He holds a degree in Computer Science and Mathematics from the University of Texas at Austin. His areas of expertise include Playback and Solution Implementation best practices for BPM Services. Fahad has written extensively on methodology, design patterns, and best practices for BPM Services.

**Dennis Parrott** is an Agile BPM Project Manager for IBM UK. He has 5 years of experience in BPM development, analysis, and project management, and has 20 years of experience in the software industry. Dennis attended the University of Warwick UK and has a Bachelor's degree in Computer Systems Engineering and a Master's degree in Parallel Computes and Computation. Dennis has developed many community assets and written extensively on design patterns, best practices, and methodology.

**Wim Peeters** is a BPM Client Technical Professional for IBM BeNelux and has been with IBM since 2001. He has 3 years of experience working with BPM and 11 years of experience in software development. Wim holds a degree in Engineering from groepT, Belgium. His areas of expertise include Java, Java Platform, Enterprise Edition, XML, SOA, IBM DataPower®, and IBM Business Process Manager.

**Jonas Zahn** is a BPM Solution Architect for IBM North America. He has six years of experience working with BPM methods and technology and more than 10 years of experience in software development. Jonas holds a degree in Civil Engineering from the University of Wisconsin at Madison. His areas of expertise include business process modeling, analysis, optimization, and implementation. Other areas of expertise include BPM program planning and initiation. Jonas has written extensively on best practices in BPM modeling and implementation.

Thanks to the following people for their contributions to this project:

- Martin Keen, IBM Redbooks Project Leader
- Phil Gilbert, Vice President, IBM Business Process and Decision Management
- Lucas Maetzschker, University of Corporate Education Mannheim
- John Reynolds, IBM Business Process Manager Product Manager
- Andrew Cannon, Business Transformation Consultant
- Marc Smith, Product Management - WebSphere / Lombardi BPM
- Christopher Walk, Senior Product Manager, IBM Business Process and Decision Management
- Jordan Krane, Senior Managing Consultant, IBM Global Business Services®
- Sean Pizel, Senior Consultant, IBM Business Process Manager Delivery
- Joshua King, IBM Business Process Manager Program Manager
- Brian French, Salient Process LLC

# Now you can become a published author, too!

Here’s an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

**ibm.com**/redbooks/residencies.html

## Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- Use the online **Contact us** review Redbooks form found at:

  **ibm.com**/redbooks

- Send your comments in an email to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

## Stay connected to IBM Redbooks

- Find us on Facebook:

  http://www.facebook.com/IBMRedbooks

- Follow us on Twitter:

  http://twitter.com/ibmredbooks

- Look for us on LinkedIn:

  http://www.linkedin.com/groups?home=&gid=2130806

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

  https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm

- Stay current on recent Redbooks publications with RSS Feeds:

http://www.redbooks.ibm.com/rss.html

# Summary of changes

This section describes the technical changes made in this edition of the book and in previous editions. This edition might also include minor corrections and editorial changes that are not identified.

Summary of Changes
for SG24-7973-01
for Scaling BPM Adoption: From Project to Program with IBM Business Process Manager
as created or updated on October 4, 2012.

## March 2012, Second Edition

This revision reflects the addition, deletion, or modification of new and changed information described below.

### New information

- Added Chapter 4, “Planning a BPM project” on page 97

# 1

# Introduction

Your first Business Process Management (BPM) project is a crucial first step on your BPM journey. Key decisions made during the early projects influence the next BPM projects. These project outcomes influence the success of an enterprise-wide BPM program. It is important to begin this journey with a philosophy of change that impacts not only your technology but also your corporate culture. Embracing a philosophy of change allows you to avoid common pitfalls that lead to failed BPM projects and, ultimately, poor BPM adoption.

You might be reading this book having already completed one or two BPM projects with varied success. Maybe you are starting your first BPM project. In either case, this book describes the methodology and best practices that lead to a successful project and how to use that success to scale to enterprise-wide BPM adoption. Successfully scaling BPM adoption from a handful of projects to a BPM program requires that we first succeed with at least one BPM project.

The intended audience for this book includes all people who participate in the discovery, planning, delivery, deployment, and continuous improvement activities for a business process. These roles include process owners, process participants, subject matter experts (SMEs) from the operational business, and technologists responsible for delivery, including BPM analysts, BPM solution architects, BPM administrators, and BPM developers. Delegates from both business and technology participate in BPM governance in a BPM center of excellence (COE).

While the BPM journey never ends, a significant milestone achievement for your enterprise is the creation of a cross-functional multilateral BPM center of excellence. As a governing body committed to continuous improvement, the COE defines and refines how your organization identifies, assesses, discovers, plans, implements, deploys, and manages your business process (Figure 1-1).

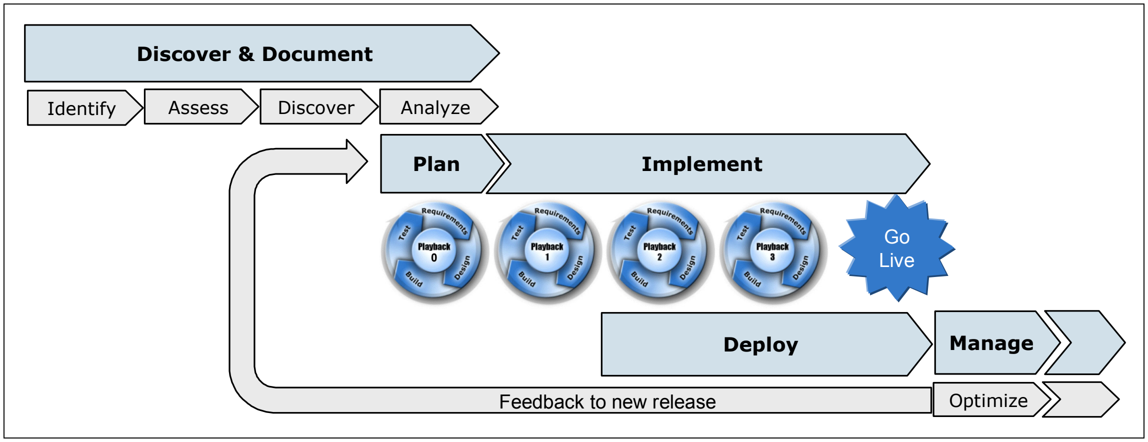


*Figure 1-1 Lifecycle of a business process*

This introductory chapter sets the stage for this book with an explanation of business process management and what it means to *manage* a process. We briefly introduce IBM Business Process Manager V7.5 to provide context for the later chapters that discover, plan, implement, and deploy the example scenario described in Chapter 2, “Business scenario used in this book” on page 27. Following the importance of your first successful BPM project, we describe the BPM journey that takes you from that successful project to a successful program. This chapter concludes with an outline of the chapters in this book organized by the lifecycle of a business process from early identification of the process through discovery, implementation, deployment, and continuous improvement of that process through management and governance.

# 1.1 Business process management

BPM is a comprehensive management approach to continuously improving your business processes. BPM is more than *workflow automation*. BPM promotes effectiveness and efficiency in your business processes by using measurable business value to align all projects with corporate strategies. BPM relies on an incremental delivery methodology that creates *process visibility*, enabling *process control* in your business. A BPM initiative should deliver targeted results that directly support the strategic goals of the business. Thus, a successful BPM initiative requires close collaboration between business operations and technologists. A successful BPM program ties all business process projects to core business initiatives. BPM allows you to manage your processes and support corporate initiatives such as improving product quality, reducing time-to-market, expanding to new markets, raising customer satisfaction, and increasing profit margins.

## 1.1.1 Managed business process

A *managed process* is a business process in which stakeholders and process owners have both *visibility* into the process and *control* over the process to take corrective action for better business outcomes. With process control, you can make informed decisions to change the behavior of your process. To manage a process is to make decisions for change. These decisions for change might affect the process, but might also impact domains outside of the business process itself. As a manager, you regularly participate in decisions for change. All management decisions, at any level in the organization, can be associated with one or more of the following domains: corporate strategy, business resources, and business processes.

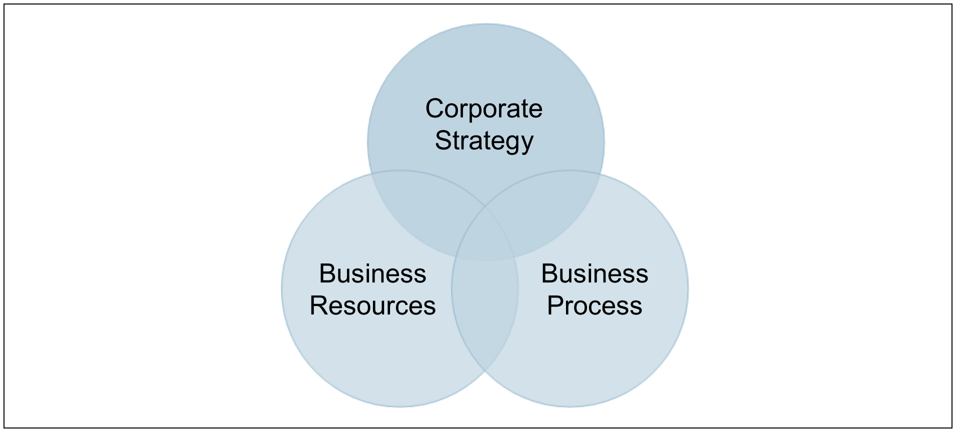


*Figure 1-2 Domains of change for a managed process*

Examples of corporate strategy decisions include entering new markets, discontinuing a product, and selling assets. You can imagine how these decisions might impact or lead to decisions in the business resource domain like hiring or training new human resources, outsourcing jobs, investing in technology, or making capital improvements to facilities. Decisions in the business process domain might include changing a decision threshold in a process flow to reduce the human workload (that is, lower the threshold) or reducing risk (that is, raising the threshold) by changing the behavior of the process. Business process decisions might also include feedback to change the process by adding or removing activities. Business process optimization decisions to streamline activities, automate human decisions with business rules, or remove bottlenecks are also good examples of decisions for change in the business process domain.

With this decision model, can you think of examples of decisions in your organization and how those decisions impact these three domains, or trigger follow-up decisions by managers within those domains? By understanding who your decision makers are, and the substance of change in those decisions, the key performance indicators (KPIs) and service level agreements (SLAs) that need to be measured and monitored within your process model become self-evident.

### 1.1.2 Business processes

There are volumes of sources that define a business process in the context of BPM. We offer the following definition of a business process from the BPMN 2.0 specification published by the Object Management Group in January 2011[1]:

> “A *process* describes a sequence or flow of *activities* in an organization with the objective of carrying out work. In BPMN a process is depicted as a graph of flow elements, which are a set of activities, events, gateways, and sequence flows that define finite execution semantics. Processes can be defined at any level from enterprise-wide processes to processes performed by a single person.”

Chapter 2, “Business scenario used in this book” on page 27, introduces the scenario used in this book as an example of a business process.

[1] Source: http://www.omg.org/spec/BPMN/2.0

Figure 1-3 shows an example of a business process definition using business process modeling notation (BPMN).

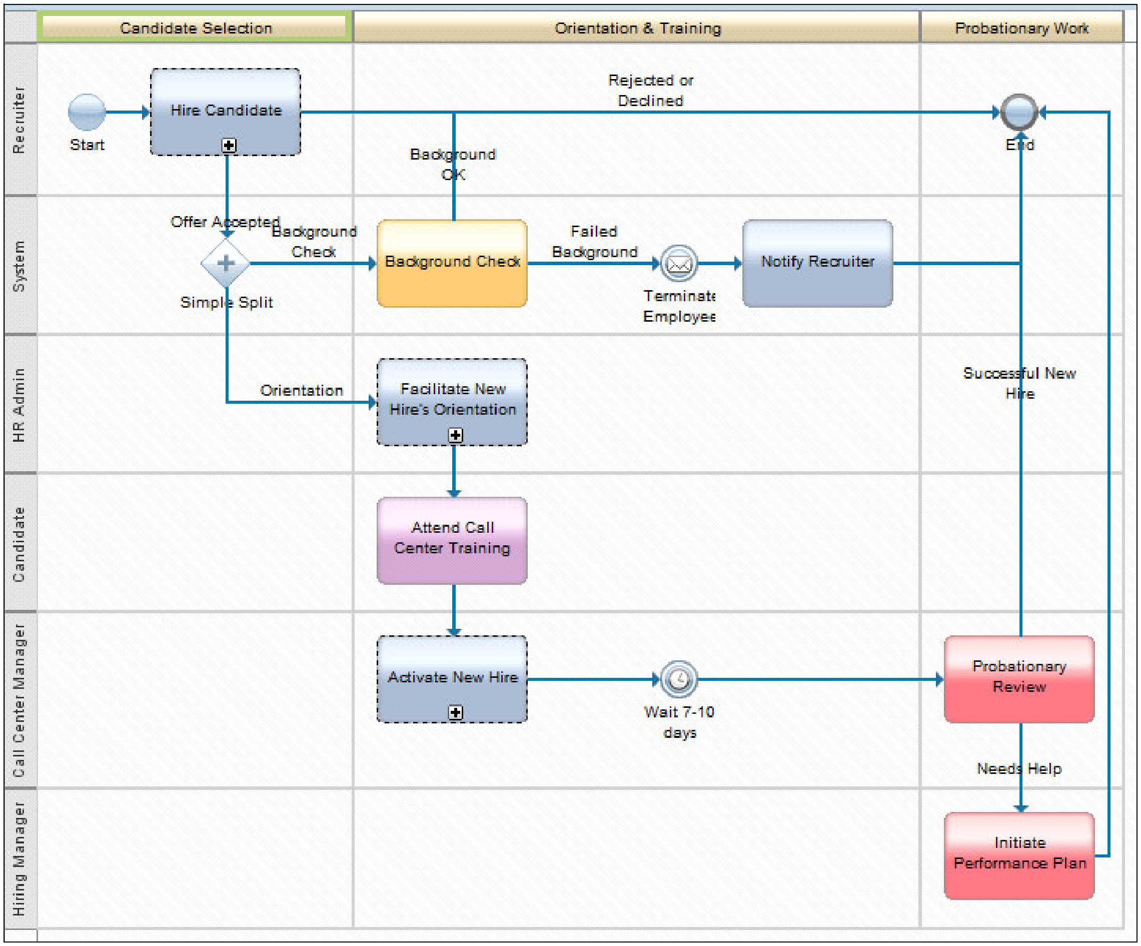


*Figure 1-3 Example of a business process definition using BPMN*

### 1.1.3 Process automation, visibility, and control

Process automation, visibility, and control are compounding elements of the business impact realized from a BPM project (Figure 1-4). Process automation immediately accrues business value by increasing efficiency, reducing errors, eliminating process variation, and removing rework for human tasks. It is important to recognize, however, that automation is not the end goal of a process implemented with BPM.

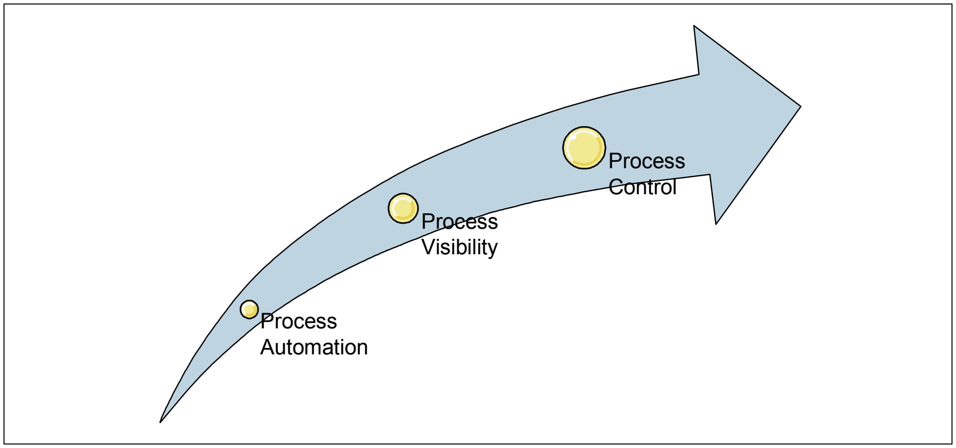


*Figure 1-4 Three elements of the compounding value of a managed process*

Process automation is not limited to activities that require system integrations. Shadow processes and *swivel chair* activities provide business value with process visibility and process control. An early release of what feels to be an incomplete process solution rift with manual swivel chair integrations often exposes new information about that business process. This new information typically exposes new opportunities and removes previous assumptions that feedback into the next release. The overall impact is shorter delivery cycles, increased business impact, and lower cost of development and maintenance (because you did not build what you did not need).

**Definition:** A *swivel chair integration* refers to a manual interface where a human user reads data from one system and keys it into another.

Process automation is only the beginning when it comes to realizing value from BPM. Process visibility allows you to see what you could not see previously. Process visibility allows you to measure what you could only estimate or guess previously. This visibility is relevant in real time (versus waiting for quarterly reports), giving you insight into the performance of your process while there is time to correct it. Process visibility also means that your process participants have perspective into the end-to-end process that is not limited to the activities assigned to them.

Figure 1-5 shows an example of how an unmanaged process suffers from poor visibility and lack of control.

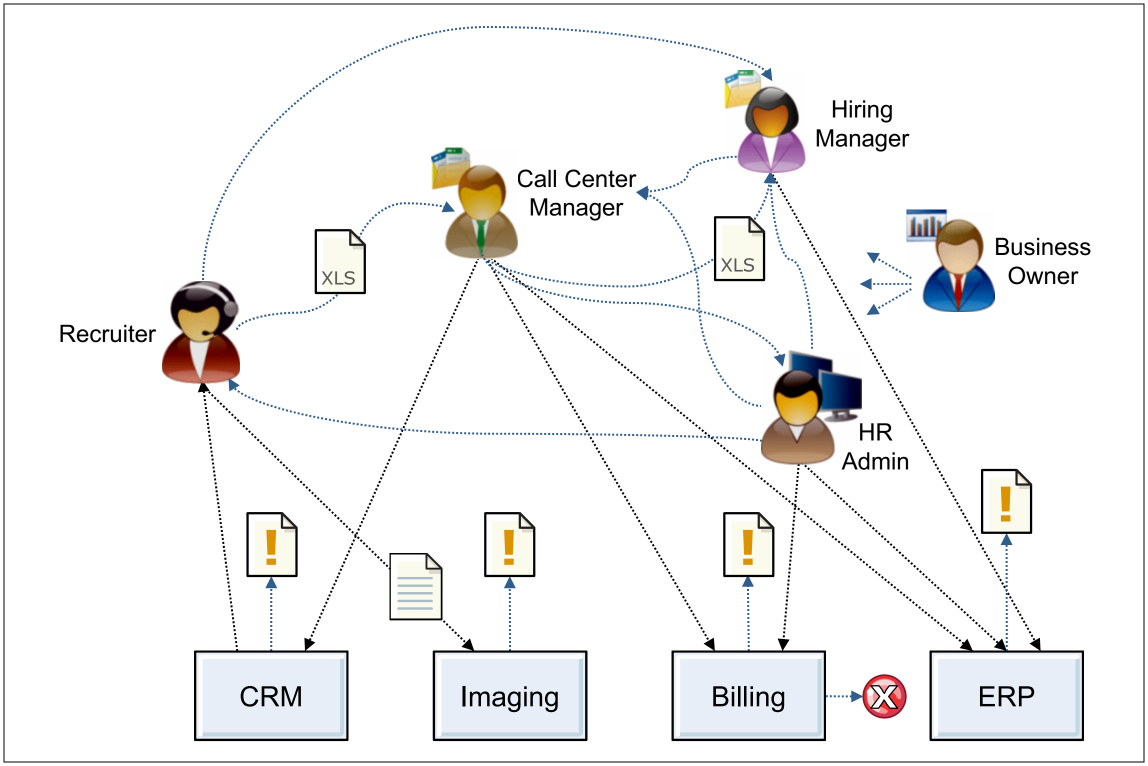


*Figure 1-5 An unmanaged process suffers from poor visibility and lack of control*

Process owners benefit from process visibility at both design time (implement and deploy) and runtime (manage and optimize) states in the lifecycle of a business process. At design time, process visibility means that all aspects within the organization know which business processes are being discovered, planned, implemented, and deployed across the enterprise. Such visibility is key to enabling continuous discovery of new opportunities to collaborate on BPM projects and share process assets in a standard way.

For process owners and corporate managers, visibility into the runtime state provides the business with the level of insight that they require to produce better outcomes and keep processes in alignment with corporate objectives. Today's complex and competitive economic environments demand a new level of business agility that can be made possible only with the automation, visibility, and control of business processes with BPM.

Figure 1-6 shows an example of a managed process with real-time visibility and control bringing order to chaos.

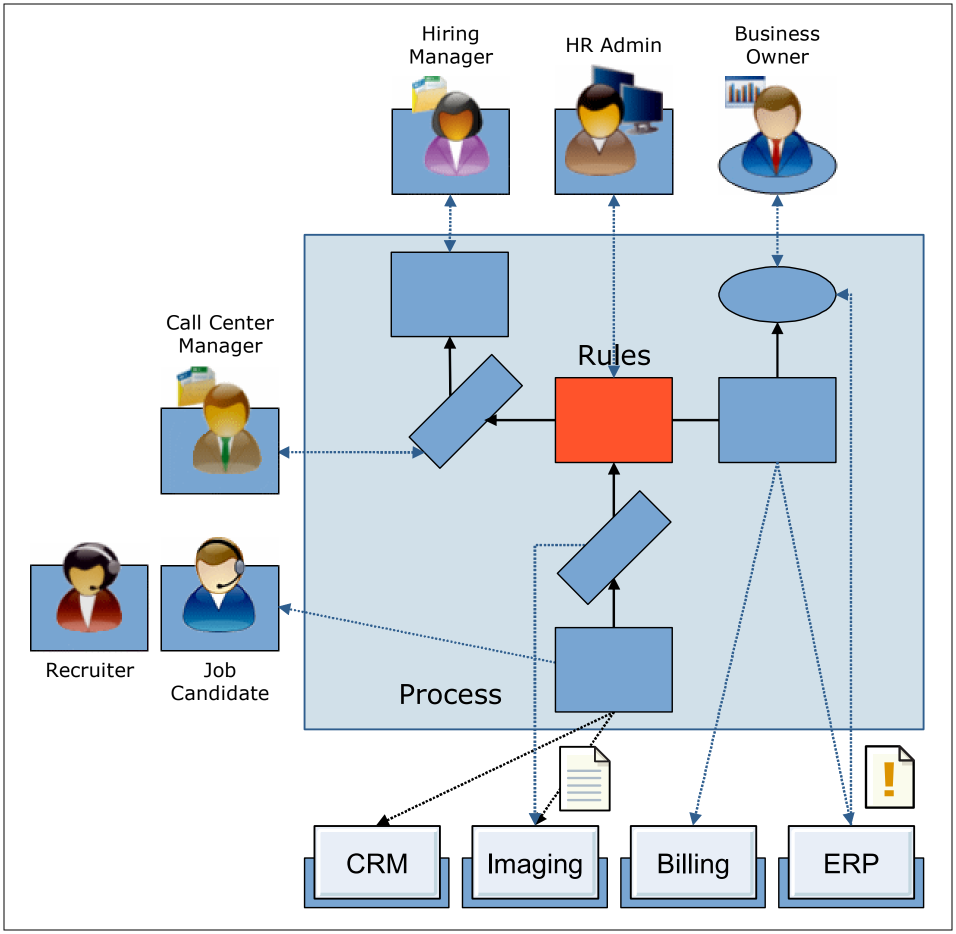


*Figure 1-6 A managed process with real-time visibility and control brings order to chaos*

Process control is what ultimately differentiates a *managed process* from an unmanaged process. Having control over your design-time and runtime business processes means acting on the insights identified and delivering a hands-on approach to process improvement that ensures that the correct people are working on the correct problems at the correct time. By default, control demands governance of not only design-time business process definitions, but also the infrastructure, the runtime business process, and all other aspects of the business process. This control ensures that processes are operating consistently and in compliance with both internal and external policies and regulations.

## 1.2 IBM Business Process Manager V7.5

IBM Business Process Manager is a comprehensive Business Process Management System (BPMS) that gives you visibility and insight to manage business processes. It scales smoothly and easily from an initial project to a full enterprise-wide program. The powerful simplicity of IBM Business Process Manager ensures success as you scale from an initial BPM project to an enterprise-wide BPM program. IBM Business Process Manager V7.5 is built to facilitate a new level of collaboration between business operations and technologists that ensures success in meeting customer demand for better business outcomes.

Figure 1-7 shows the logical architecture of IBM Business Process Manager V7.5.

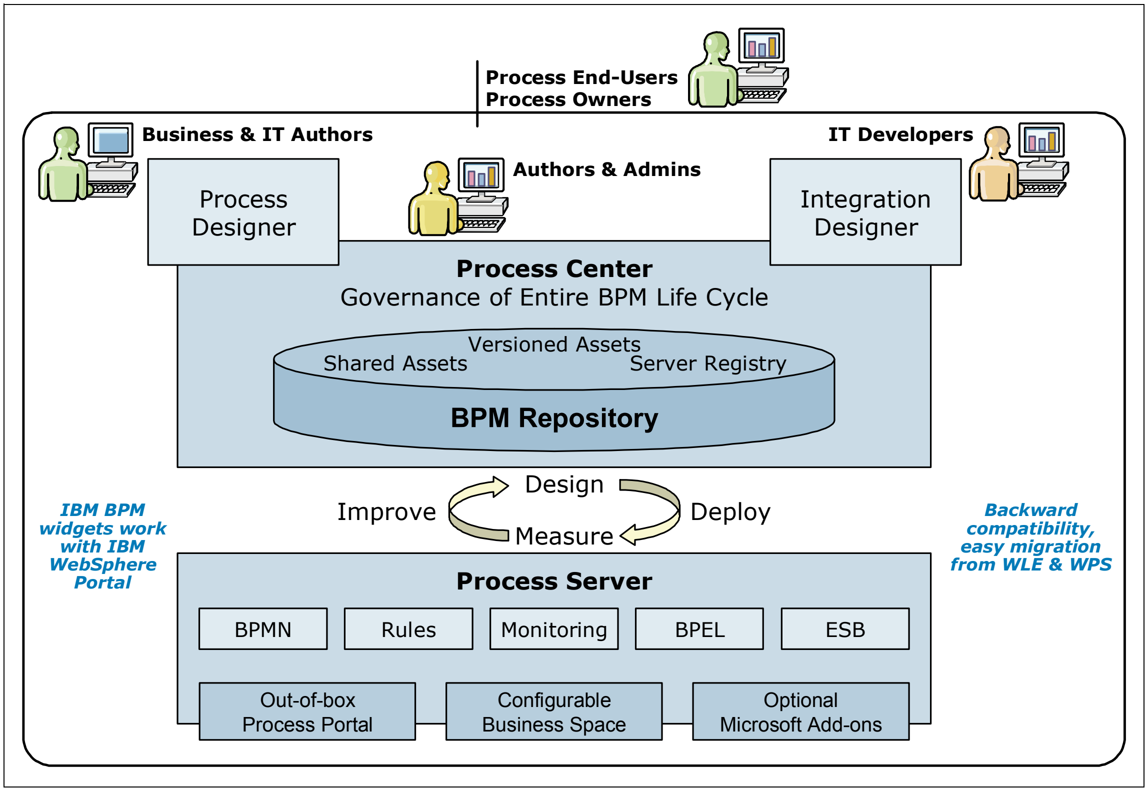


*Figure 1-7 Logical architecture of IBM Business Process Manager V7.5*

IBM Business Process Manager comes in Advanced, Standard, and Express configurations because continuous process improvement is an imperative for organizations of all sizes. For more information, see:

- IBM Business Process Manager: Optimize business processes to reduce complexity and boost productivity

  `http://www.ibm.com/software/integration/business-process-manager/#`

- IBM Business Process Manager V7.5 information center

  `http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/topic/com.ibm.wbpm.main.doc/topics/cbpm_ibpmarch.html`

The sample scenario in this book requires IBM Business Process Manager V7.5 Advanced configuration.

## 1.3 The importance of a successful BPM project

When you first embark on anything new, you need validation that you are on the correct path. Achieving an initial quick win with BPM validates your investment. This initial win also helps you populate your BPM journey with future successes that steer you toward a successful BPM program.

When a BPM project does fail, it is often due to trying to accomplish too much and too soon in the first release of the project. Others fail by trying to force BPM to fit into an existing service-oriented or API-oriented architecture. The mistake in a failed attempt at BPM occurs when architects perceive BPM as a component of, or a replacement for, SOA. While SOA is a flexible set of design principles that influence system integration, BPM is a management approach to continuous business improvement. However, BPM and SOA are not incompatible. In fact, a successful BPM program uses reusable capabilities exposed by good SOA including access to business data, execution of business rules, and implementation of business logic.

Figure 1-8 shows an example of using an early win to foster BPM adoption.

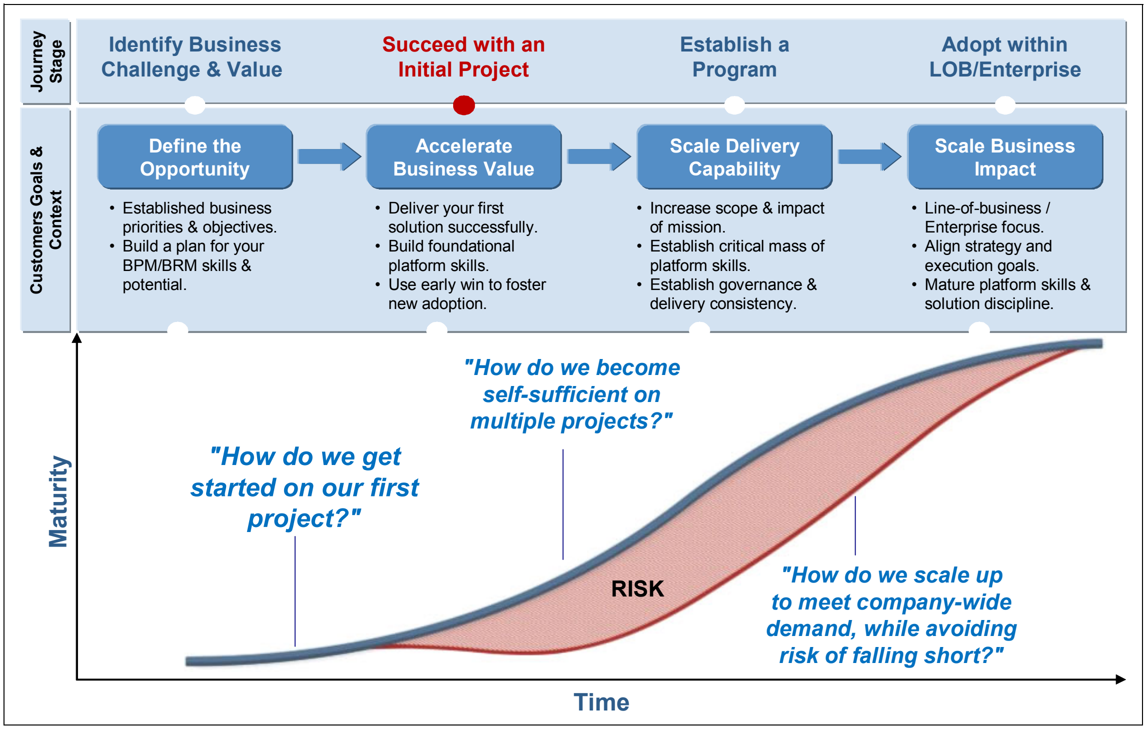

Journey Stage
Identify Business Challenge & Value
Succeed with an Initial Project
Establish a Program
Adopt within LOB/Enterprise
Customers Goals & Context
Define the Opportunity
Accelerate Business Value
Scale Delivery Capability
Scale Business Impact
• Established business priorities & objectives.
• Build a plan for your BPM/BRM skills & potential.
• Deliver your first solution successfully.
• Build foundational platform skills.
• Use early win to foster new adoption.
• Increase scope & impact of mission.
• Establish critical mass of platform skills.
• Establish governance & delivery consistency.
• Line-of-business / Enterprise focus.
• Align strategy and execution goals.
• Mature platform skills & solution discipline.
Maturity
"How do we get started on our first project?"
"How do we become self-sufficient on multiple projects?"
RISK
"How do we scale up to meet company-wide demand, while avoiding risk of falling short?"
Time

*Figure 1-8 Use an early win to foster BPM adoption*

The first successful BPM project is an opportunity to identify best practices, roles, methodology, and a baseline against which your organization can measure future success. We say *first* successful BPM project because we recognize that not all first BPM projects are successful. It is important to achieve success with a BPM project and share the delivery methods, implementation patterns, and best practices established in a successful BPM project. Making these items repeatable is the key to building a BPM program.

In the beginning, you have some flexibility with your first BPM projects. You should capitalize on this opportunity to try a new methodology for BPM rather than attempting to make BPM fit your existing methodology. Part of the reason that you are trying BPM is because you need to solve problems that your existing methods cannot.

In preparation for the process discovery phase, you create an inventory of business process candidates from which you select your first BPM projects. Select projects with a high likeliness of success. Selection criteria for your first BPM projects should include an initial assessment of:

- The business impact
- The scope and complexity of the process
- The appetite for BPM on behalf of the process owner and SMEs

Chapter 3, “Process discovery” on page 41, provides guidance on these selection criteria for your first BPM projects.

## 1.4 Moving from project to program

BPM can help you build a business that is adaptable, and adaptability in and of itself is becoming a value proposition, maybe *the* value proposition, of this first half of the 21st century. Adaptability by humans to changing conditions, although enabled by using the correct technology, is a cultural issue. Adaptability itself is a company's biggest competitive advantage. Only if you move the existing culture do you succeed at scaling BPM from a few projects to a comprehensive BPM program.

> **Transforming your enterprise:** Transforming your enterprise requires time and patience, investment in people and technology, and commitment from executive leadership, middle management, and the workforce.

An enterprise running a successful BPM program institutionalizes and makes the techniques employed to discover, implement, deploy, and manage individual BPM projects repeatable. If you recognize some of these characteristics within your organization, you are likely already running a BPM program even though you are not explicitly calling it that:

- Multiple BPM projects in progress simultaneously
- Shared infrastructure for running processes
- Shared repository for storing all process knowledge
- Shared implementation team
- Shared vocabulary and language
- Repeatable methods to identify, assess, discover, and analyze processes
- Repeatable methods to implement, deploy, and manage processes
- Shared collection of best practices
- Shared vision for the enterprise value chain
- Common goal and vision for all projects (enterprise success)
- Continuous training and enablement activities for a growing team

BPM transformation is a journey (Figure 1-9).

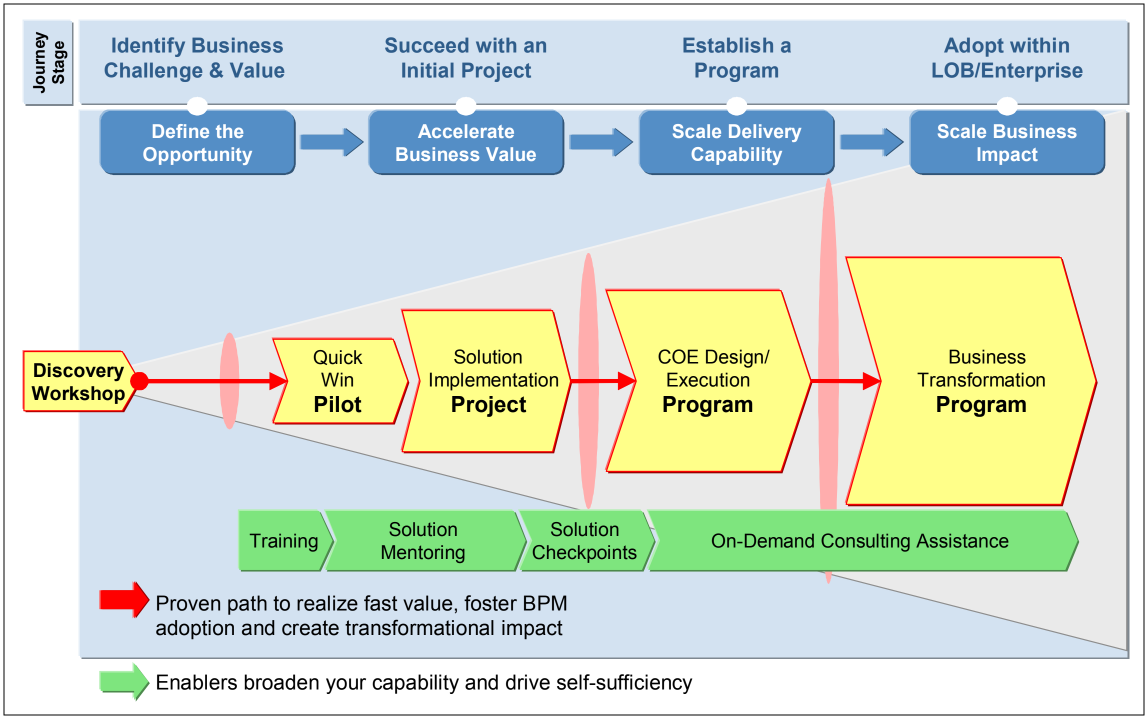

Journey Stage
Identify Business Challenge & Value
Succeed with an Initial Project
Establish a Program
Adopt within LOB/Enterprise
Define the Opportunity
Accelerate Business Value
Scale Delivery Capability
Scale Business Impact
Discovery Workshop
Quick Win Pilot
Solution Implementation Project
COE Design/ Execution Program
Business Transformation Program
Training
Solution Mentoring
Solution Checkpoints
On-Demand Consulting Assistance
Proven path to realize fast value, foster BPM adoption and create transformational impact
Enablers broaden your capability and drive self-sufficiency

*Figure 1-9 BPM transformation is a journey*

A common theme throughout this book is that BPM has a lot more to do with people than with technology. As stated recently by a BPM program director after working more than six years to establish a BPM program, “With [BPM], it’s not what you do that is so different. It’s how you do it. That is the big difference.” Investment in skills development is as important, if not more important, as the investment in technology. A BPM enablement plan with skills development activities helps build foundational platform skills. These skills are followed by continued investment to establish critical mass of platform skills in your organization. These actions are the defining measure for the success of your BPM program. Without the people, there is no program.

## 1.4.1 BPM enablement through skills development

Improving your business with BPM and transforming your organization requires a cultural change, and more specifically this transformation requires a *culture of change*. It is human nature to resist change. The people chosen to lead and deliver the first BPM projects, if successful, will develop a new perspective of what change means and how important change is to the organization. Your team will learn to embrace change and begin adapting this new perspective to your organization.

**Culture of change:** Success in BPM depends on creating a *culture of change*.

As your BPM team develops and all roles consistently trace their work to business value, you begin to see the effects of a *process-driven* culture. This transformation begins with a collaborative and comprehensive perspective of a *business process* and the business value derived from that process. The value proposition is the conversation starter. It is the premise in the business case used to justify continued efforts in discovery, planning, implementation, and continuous improvement activities. It is the way that we measure success and the impact of change, and it is the basis of your ROI for BPM.

### Skills development must be planned

It takes time to change corporate culture, and those organizations who are successful do so incrementally. A successful BPM program depends on a wide range of skill sets from all segments of your organization, including process participants, process owners, analysts, developers, program administrators, architects, and administrators. Developing these skill sets mandates a formal enablement plan to include formal training, mentoring, and a mix of on-demand assistance and staff augmentation with IBM professional services that suits the needs of your BPM projects.

### Skills development requires experience

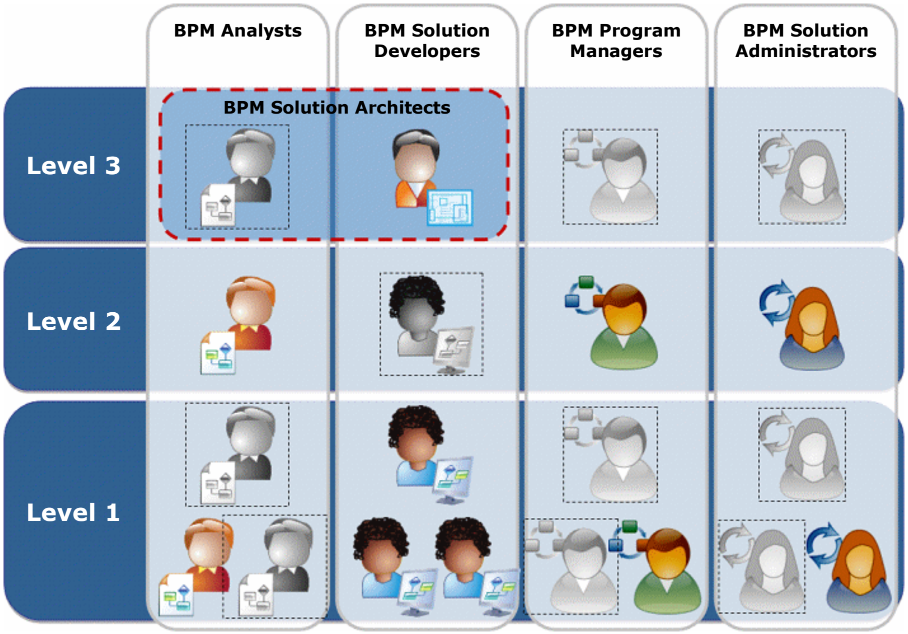
Training and mentoring gets your people started with BPM. Becoming an expert in any job function requires years of experience and reflection on lessons learned deploying and *managing* different processes within your organization. Experience starts with one successful project and then grows as you move on to larger projects, ultimately formalizing a BPM program and managing dozens of process-focused, value-driven business process applications.

## Skills development brings your people together

As your organization transforms and your people adopt BPM, they speak a common language across the functional boundary that often separates operational business people from technology people. A similar boundary divides working class process participants from executives and middle managers. In BPM, all people accept responsibility for corporate objectives and the business value that their business processes contribute to corporate objectives.

## Skills development should be role-oriented

The roles and responsibilities of the people working on BPM projects and programs have characteristics that are unique to BPM. Each role in a BPM project carries different responsibilities, and each is imperative to the success of a BPM project. Your skills development plan needs to include a role-based curriculum and should not be a “one size fits all” training class. This section provides you with a high-level skills road map to help you select and educate your BPM team (Figure 1-10).



Sufficient skills development prevents slow adoption, lost value, and program failure.

*Figure 1-10 Education is essential to self-sufficiency in BPM*

### BPM analyst

The role of the BPM analyst is to analyze and help document the business process and its key performance indicators (KPIs), service level agreements (SLAs), business reports, process metrics, and process diagrams. A BPM analyst leads collaborative workshops that include corporate executives, process owners, and subject matter experts in process discovery activities such as process identification and inventory, process assessment, process discovery, and process analysis.

Responsibilities and skills of an experienced BPM analyst include:

- Maintaining an inventory of business processes
- Assessing business processes and helping select processes for further discovery
- Discovering and documenting business process diagrams
- Creating a business case to justify implementing a process
- Helping plan strategic initiatives in BPM
- Helping plan and establish a BPM program
- Leading process optimization workshops with process owners

### BPM developer

A BPM developer collaborates with BPM analysts, process owners, and subject matter experts to gain a clear understanding of the business process. The BPM developers design and implement solutions that include process applications and toolkits. For a BPM developer, a BPM project encompasses the following topics and activities:

- What the justification for the project was (why this money is being spent)
- What was sold
- End-to-end process understanding
- Reporting requirements
- Data and system requirements
- User interfaces
- Roles, task routing, and user repository requirements
- Testing methods
- Issue tracking
- Infrastructure requirements handoff
- Process solution scope review

After the business problem is understood and the solution architecture is developed, the BPM Developer implements the process and the business reports based on input from the BPM analyst. The BPM developer also collaborates with BPM solution architects and BPM solution administrators on hardware metrics and integration metrics.

Experienced BPM developers can advise executive leadership on BPM adoption. They are able to work on multiple customer projects simultaneously in an advisory and leadership capacity, bringing a voice of compelling experience to clients. BPM developers who attain Level 3 experience and certification often transition to the BPM solution architect role.

### BPM program manager

The BPM program manager is an expert in BPM methodology and is the project leader in resource staffing, planning, estimation, implementation, deployment, readiness, and enablement. The BPM program manager is responsible for the cadence in an iterative delivery including planning, committing, and validating development iterations with process owners and key stakeholders. The BPM program manager helps organize the team, engages process owners and process participants for development workshops, coordinates playbacks, and provides real-time visibility and oversight for the project. A BPM program manager responds to and negotiates issues that occur during projects, often collaborating with the project sponsors to guarantee the projects success and good messaging.

### BPM solution architect

A BPM solution architect is often a senior BPM developer with five or more years of experience in BPM who has participated in the discovery, analysis, planning, implementation, deployment, and continuous improvement of multiple business process solutions. The role of the BPM solution architect includes designing business process applications based on both business and technical inputs. Persons in this role can lead teams and perform both BPM analyst and advanced BPM developer responsibilities.

### BPM solution administrator

A BPM solution administrator is responsible for managing access control to the design-time process assets (who can change what in process assets) and deployments of processes to runtime environments. Deployment management includes control of which runtime environments are a part of the process center and control over which snapshots are deployed, activated, and archived.

In addition, BPM solution administrators also have responsibility for the values of exposed process variables, which can govern business thresholds.

### BPM process owner

The process owner, typically the manager of the business unit or department that owns the process, is responsible for the adoption and overall success of the business process. The process owner is the ultimate decider about features and functionality and scope negotiations during iteration planning, commitment, and acceptance. The process owner participates almost daily in all phases of the project, from early discovery to post-deployment process management and optimization.

### BPM process participant (SME)

Process participants are typically team leads or expert users who provide practical and deep insight into the details of the business process. These SMEs collaborate with BPM analysts and BPM developers during the discovery, planning, and implementation phases.

**SME participation:** Do not underestimate the time required from SMEs during the project. Plan for SMEs to be available roughly 80% of the time during the first few iterations (Playback Zero and Playback One) and 25% to 50% thereafter. SME time (man hours) is roughly 1/3 of the total time (man hours) spent on analysis, planning, and implementation. Put another way, one full-time SME paired with one full-time BPM analyst and two full-time BPM developers is a good fit for a BPM delivery team.

## 1.4.2 The importance of a BPM program

BPM is most effective and valuable when deployed at an enterprise level integrating all processes with the value chain and extending the reach and impact of shared processes and process assets. In the beginning of your BPM journey, you use BPM tools and methods at a tactical or departmental level to solve specific contemporary business problems. This way, new tools and methods are funded and evaluated. The challenge comes in using a departmental success in BPM to scale your BPM initiatives to an enterprise-wide BPM program. This transformational journey for your enterprise is the heart and soul of IBM Business Process Manager.

**Culture change:** Only if you move the existing culture do you succeed at scaling BPM from a few projects to a comprehensive BPM program.

No process is atomic. Every process hands off to, initiates, or is affected by other processes. Your inventory of processes yields additional wisdom after you reach a critical mass of deployed processes. With process visibility and traceability to the enterprise value chain, you realize compounded business value with each new process application and align that value to corporate strategy. You realize economies of scale measured by faster development, more frequent deployments, and a growing team of experts.

Failure to establish a BPM program results in tribal use of BPM tools, leaving significant value on the table. You do not realize the economies of scale that a BPM program promises. Without a program, your distributed BPM teams do not follow consistent delivery methods. Teams do not take advantage of shared resources, and you do not realize faster development, more frequent deployments, or a growing team of experts. Establishing a BPM program is an important milestone on your BPM journey.

Figure 1-11 provides an example case study of the value of BPM.

Call Center C is in the business of providing call center services to companies who outsource call center operations for both incoming and outgoing calls. The core business of Call Center C is producing teams of highly effective call center representatives. Therefore, improving the process to hire, train, start, and retain new call center representatives scales the core business of Call Center C.

However, before using BPM methods to discover, analyze, and implement this process, hiring, training, starting, and retaining call center representatives were separate functions measured differently and sometimes in competing ways. Such is the case by measuring the recruiter's performance by the number of candidates submitted without factoring in the number of offers extended and offers accepted by the candidates. Such is also the case when measuring the effectiveness of the hiring process by number of new hires per week without taking into account the performance, retention, and cost of those new hires.

Combining these functions into a single cross-departmental business process with a consistent measure of performance aligns all participants with the corporate objective to produce high-performing call center teams.

*Figure 1-11 An example case study on the value of BPM*

### 1.4.3 Creating a BPM program by getting your projects noticed

We have made the case for the importance a BPM program and that a program builds from the successes of your first BPM projects. Now we show you how to move from one or two quick win process deployments to a number of different, inter-related process deployments that have a measurable impact on your enterprise value chain.

It should be no surprise that the same principles used to improve a business process can be used to improve a BPM program:

- Measurable business impact
- Incremental delivery
- Transparency and visibility
- Control over your business

*Measure* the value and impact of BPM by proving business value in short iterative cycles. *Market* that value early and often by providing visibility into that value for stakeholders and management. *Repeat* these procedures and prove that the value of BPM is not unique to a single process, department, or delivery team. You can perform all of these processes by using a shared enterprise-ready BPM platform to implement, deploy, and manage your business processes with IBM Business Process Manager V7.5.

#### Measuring your success: Proving business value

*Prove business value first.* The focus on every project must be on measurable business value. That business value should trace to corporate objectives. All projects should be measured by *business value* and not *functionality*. For many technologists, this situation is a paradigm shift in thinking about how information technology is delivered to the customer (the business). For years, we learned many creative ways to measure software:

- Scope: Number of features
- Test coverage: Thousands of lines of code (KSL)
- Quality assurance: Count of defects per KSL

**Measuring quality:** In BPM, the only measure of quality is adoption, and adoption immediately follows business value.

*Do not skip process analysis.* Process discovery and analysis are not the same as gathering requirements and writing a requirements specification. Culminating with playback zero as the last stage of process discovery comes process analysis. With an in-depth shared model of the as-is business process, you have the opportunity to analyze and create a to-be process. You might recognize common process patterns, bottlenecks, and opportunities to remove, reassign, or regroup activities. This situation might be the first time that anyone in your organization has a comprehensive view of the end-to-end process. Do not miss the opportunity to find value with a few days of analysis workshops.

*Make BPM about productivity* and *visibility.* Most people perceive BPM as another workflow automation tool. Some see the opportunity to improve process flow with up-front process re-engineering. But the real value comes in the visibility made possible by defining metrics, key performance indicators (KPIs), and service level agreements (SLAs). Define these items first, not later. Document key metrics during process discovery, not after implementation starts. Do not de-scope the metrics. Visibility is critical to improvement.

*Deliver early and often,* never one and done. BPM is about continuous process improvement measured by incremental business value attained in short cycles. For many business owners, this situation requires a paradigm shift from traditional big-bang deployments. Negotiate scope by business value. Carefully consider the value of time, because in this case *less* really is *more*. An early release in weeks or months instead of years starts returning business value immediately *and* provides new information and direction for the next release. After a series of incremental releases, you are more likely to produce a better business outcome without having the risk of building what you did not need. The result is faster cash returns on your investment.

## Marketing your success: Making it visible

*Establish process owners.* Each business process has a single process owner. If there is no business owner, there is no process, and effort on the process should halt with the earliest phase in discovery: process identification. If multiple process owners are named, then find their manager and trace up the management hierarchy until you find a single name. The process owner markets the success of the business process and make it visible in the organization.

*Fund to business value.* Chartering a new project that is funded to the business value realized by the project keeps all eyes, particularly executive management, focused on value. A project charter that funds a project for the first release, a requirement specification, or a list of features, shifts the definition of success and the focus to the artifacts delivered and away from the business value created by the project.

*Tie BPM project value to corporate strategy.* If you fund projects to business value that is aligned with corporate strategy, you can be assured that executive management notices.

*Market your work.* Create regular internal communication about progress. Make visible the metrics (KPIs and SLAs) that you are using to track project success. Demonstrate value and use videos, wikis, portals, reports, and so on, to broadcast the success of your new business processes. Use testimonials to illustrate impact on the daily routines of process participants. In our experience, people do not really understand the effect of BPM until they see it. Do not take anything for granted. Show people the simplest of scoreboards (that is, My Team Performance) that a manager can look at every day to see who is doing what or the Process Performance scoreboard and drill down to find where the bottlenecks are.

## Repeating your success: Making it scale

*Build a complete team.* Scaling BPM adoption means more projects. More projects means more people (and not just developers). The success of your program depends on the right mix of team resources. Make BPM enablement for all roles a priority in your program. Budget adequately to train, mentor, and continuously reinforce the skills needed in your people.

*Make self-sufficiency a priority.* Do not mix self-sufficiency with tight deadlines. Plan your project schedules with time to adequately develop skills in your people. Projects with tight deadlines due to external factors (that is, regulatory compliance) should not rely on developing teams. Use expert partners or staff augmentation instead. Do not allocate partial human beings. Making self-sufficiency a priority means fully allocating people to BPM and fully committing to developing their skills for success in BPM.

*Take time to delivery value.* A project longer than 90 days is not a failure. While one goal is to shorten delivery cycles within your organization, your projects are measured by value delivered and not by project duration. In your early projects, the schedule needs to accommodate time for training and mentoring. This time is an investment in your people, and your people are key to scaling BPM adoption.

*Force collaboration.* Repeatable iterative development relies on the daily interactions of people. Your project delivery teams, including BPM analysts, BPM developers, the BPM program manager, the process owner, and subject matter experts need to interact daily. For this reason, collocation is suggested, especially for your first projects. When selecting your early BPM projects, select projects where collocation is easy. Do not postpone or skip playbacks. Playbacks are essential tools that not only validate process deliverables, but also foster collaborative development.

**Manifesto for Agile Software Development:** The manifesto (http://agilemanifesto.org/) is:

> "We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:
>
> – *Individuals and interactions* over processes and tools
> – *Working software* over comprehensive documentation
> – *Customer collaboration* over contract negotiation
> – *Responding to change* over following a plan"

## 1.5 How this book is organized

The next six chapters in this book are organized by the phases in the lifecycle of a BPM project. Each chapter is a procedural guide and reference that illustrates the activities in each lifecycle phase with examples from the business scenario. Additional materials, including the fully implemented sample process, are referenced in the appendix.

Figure 1-12 shows the lifecycle of a business process.

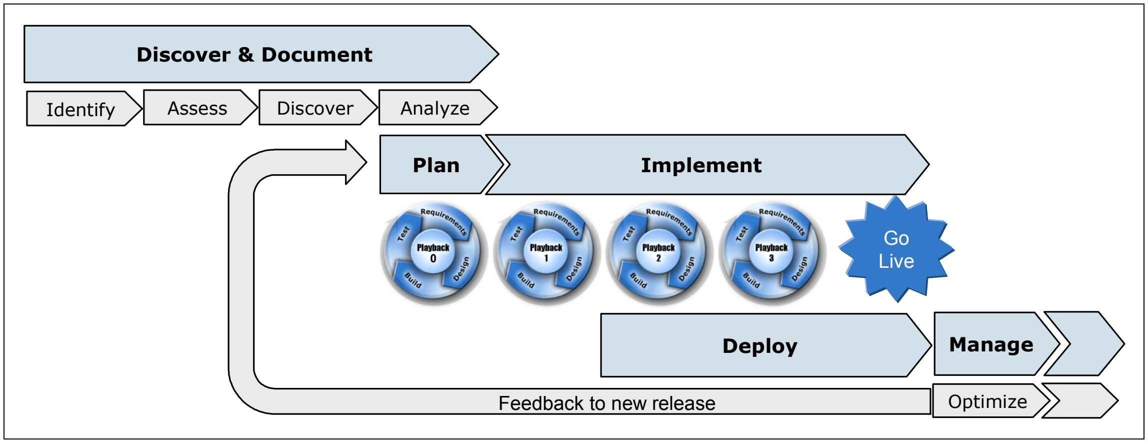


*Figure 1-12 Lifecycle of a business process*

This book contains the following chapters:

- Chapter 2, "Business scenario used in this book" on page 27
- Chapter 3, "Process discovery" on page 41
- Chapter 4, "Planning a BPM project" on page 97
- Chapter 5, "Implementing a BPM project" on page 161

- Chapter 6, “Deploying a process” on page 183
- Chapter 7, “Managing a process” on page 199
- Chapter 8, “Business process governance” on page 225

# 2

# Business scenario used in this book

This chapter describes the business scenario that we use within this book to demonstrate how to discover, document, implement, and manage business processes in IBM Business Process Manager.

This chapter introduces the scenario, describes the company and people involved in this scenario, and introduces the business processes.

The business scenario in this book covers the hiring and onboarding of new call center representatives.

## 2.1 The company

*Call Center Company C* is a fictitious company that provides call center services to its customers. The company is based in the United States and operates various call centers in the US, India, and China.

**Fictional company:** We use a fictional company to illustrate examples throughout this book This fictional company is used for instructional purposes only.

Due to an interesting business model, Call Center Company C observes a large interest in its services and an increasing number of requests from existing and new customers. Call Center Company C offers the outsourcing of call centers and expects large clients to sign a contract with them soon. If one or more of these significant deals is made, the company needs to onboard around 500 call center representatives in a short time frame.

With its current staff, Call Center Company C cannot serve the huge number of requests and needs to hire a significant amount of call center representatives. The process for hiring and onboarding new call center employees is currently manual and is not documented. For Call Center Company C, it is important to make the hiring of new call center representatives as efficient as possible and to get the new employees productive in a short time. Because hiring and onboarding new people is a core part of a call center company, the improvement of this core process results in a direct improvement of the overall value chain.

## 2.2 The people

Call Center Company C employs a number of people with different roles. The spectrum of roles ranges from administrative and technical roles to business-focused roles, including call center representatives.

This section lists all the roles for people who are part of the scenario and who are involved directly in the onboarding process. The onboarding process of call center representatives involves employees of Call Center Company C and also external people.

## Recruiter

Call Center Company C needs to hire many new call center representatives. For this purpose, they use professional *recruiters* to identify potential job candidates and to manage the communication with the job candidates.

## Job candidate

All people who apply for a job at Call Center Company C and all people who are identified by the recruiters take the role of a *job candidate*. Job candidates participate in a job interview, need to perform a test, and might be offered a call center job thereafter. If a job candidate accepts the job offer, the role changes from job candidate to *new hire*.

## New hire

*New hires* are people who successfully passed the hiring process in Call Center Company C and who are working as a call center representative. New hires must attend a call center training before they start working as a call center representative. The training helps them to become efficient in a short time.

## Hiring manager

The *hiring manager* plays an important role in the overall scenario. The hiring manager performs the interviews with the job candidates, decides whether to hire a job candidate, negotiates the contract, and manages the communication with the recruiters. If the probationary work period of a new employee was not successful, the hiring manager creates a performance plan for that employee.

## Call center manager

The *call center manager* is participating in the onboarding of new hires. The call center manager plans the work schedule for new employees and activates their profiles in the system. The call center manager also runs a probationary review with the new employee after 7 to 10 days of work.

#### Human resources (HR) administrator

The *HR administrator* welcomes new employees in the company and helps them to get started with their first days on the job. The HR administrator makes sure that the new employees attend the new hire orientation, fill out various HR forms, and receive their equipment. The HR administrator also enters the employee information into the employee database.

## 2.3 The process

At Call Center Company C, the process for hiring and onboarding new call center employees was manual and differed slightly depending on the persons who handled the case. The CEO of Call Center Company C commissioned a team of business analysts to redefine the hiring and onboarding process using IBM Blueworks Live™ (https://blueworkslive.com). The CEO decided to use IBM Business Process Manager for the implementation of this important business process to increase the process efficiency and to be able to handle the upcoming business.

In this section, we describe briefly the high-level process that the Business Process Management (BPM) team of Call Center Company C implements. We describe the aspects of the realization of this project throughout this book.

### 2.3.1 The milestones and activities

In this section, we address the milestones and activities of the process.

**Color-coding in figures:** We use the following color-coding for the activities listed in the figures throughout this book:

- Red

  Activities highlighted in red include uncertainty and require further investigation.

- Green

  Activities highlighted in green are activities that require a direct system integration.

- Lavender

  Activities highlighted in lavender are out of the scope of this book and are implemented as swivel-chair integration. This situation means that user needs are presented with a user interface with instructions and optionally business data and needs to go to another system and reenter the data. This solution is often a Release 1 solution that can be replaced in a later release when the appropriate system integration is made available.

- Yellow

  Activities highlighted in yellow do not represent a business activity in process, but represent a coach (a workflow step).

- Orange

  Activities highlighted in orange are activities that contain advanced business logic or integration services.

- Bright blue

  Activities highlighted in bright blue are activities that contain decisions or rules.

- Gray

  Activities highlighted in gray represent external activities that are not part of the implemented business process.

- Default blue

  Activities that do not fall in any of the other categories are highlighted in the default color.

Figure 2-1 shows the discovery map with the activities and milestones of the call center representative onboarding process.

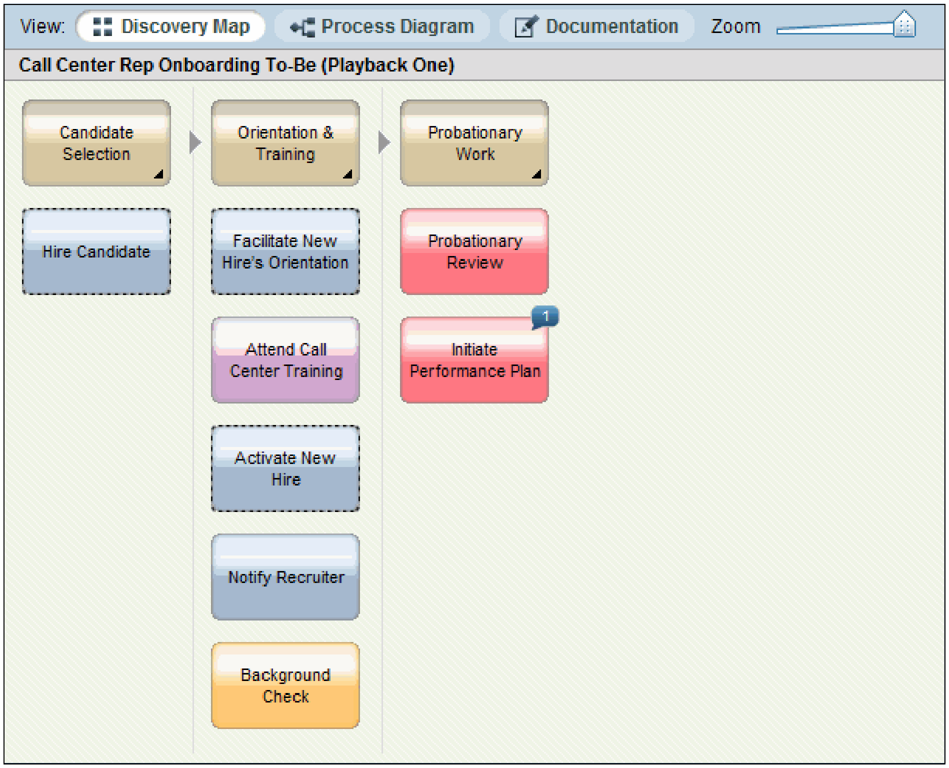


*Figure 2-1 Business scenario: Activity overview*

The activities that need to be performed to hire and onboard a new call center employee begin with the selection of job candidates. The selection of a candidate involves several steps that are not shown in Figure 2-1. These steps include a screening test that needs to be taken by the job candidate, a job interview, the hiring decision, and in case of success, the making of a job offer. The hire candidate activity is modeled as a subprocess (indicated by the dotted lines around the activity) and represents the only activity that is included in the *first milestone (candidate selection)*.

Assuming that a candidate was selected based on the interview and the test results and was made an offer, then a background check for the person is initiated. Only candidates that pass the background check are hired. Besides the background check, the *second milestone (training)* contains additional activities that include orientation activities in the first days of employment, a call center training, and the activation of a new hire in the systems. As shown in Figure 2-1 on page 32, the second milestone contains two subprocess activities:

- Facilitate new hire's orientation.
- Activate new hire.

After all of these activities are complete, the call center employee is ready to start working. To help him during the first days, a mentor is assigned to him and a probationary review after 7 to 10 days of work is performed to ensure that there are no open questions and that the new employee can perform his work efficiently. If there are problems with the work efficiency of the new hire, then a performance plan is created to resolve this situation. These activities are included in the *third milestone (probationary work)*.

### 2.3.2 The call center representative onboarding process

Figure 2-2 shows the actual onboarding process. It shows the activities of the discovery map, places them in swimlanes, and defines the sequence. The swimlanes are assigned to the roles that perform the different activities.

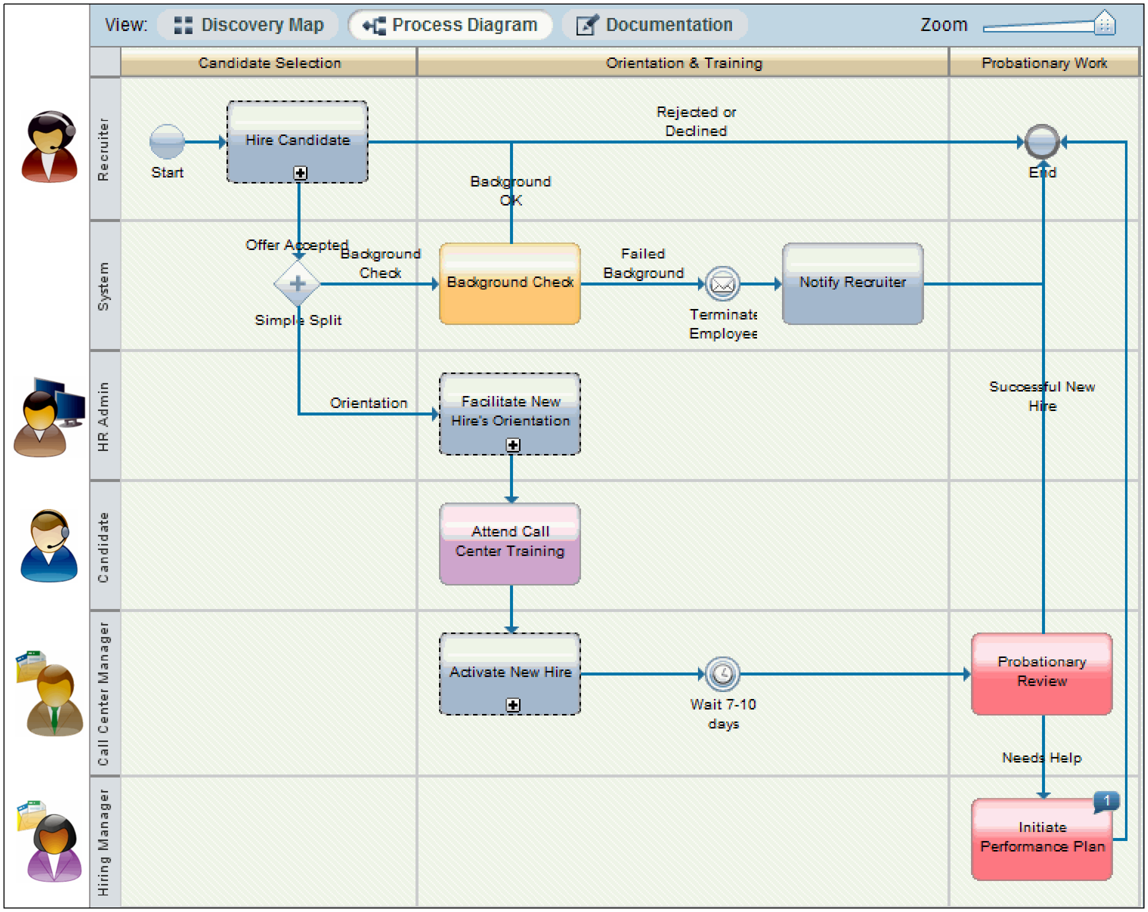


*Figure 2-2 Business scenario: Call center representative onboarding process overview*

The business process starts with the *hire candidate* subprocess, in which different activities are contained. The *hire candidate* activity can return with:

- The rejection of a candidate
- The decline of a job offer
- The acceptance of a job offer

In cases 1 and 2, the process ends. Only in the successful cases are further activities performed. This activities include a *background check* and the *facilitate new hire's orientation* activity. The background check involves a criminal background check, a credit check, and a social media background check. The results of the three checks influence the pass versus fail decision on the background check. If a candidate does not pass the background check, a termination process is initiated (*terminate employee*) and the recruiter is informed (*notify recruite*r).

> **Parallel execution:** As you can see in Figure 2-2, most of the activities require a specific order and are performed sequentially. There is only one place where activities are in parallel and where a split is used. This place is the split before the background check and facilitate new hire's orientation activity.

After completing the new hire orientation phase, the new employee attends several call center courses to get familiar with the procedures, the equipment, and the call floor etiquette (*attend call center training*). The new hire is now ready to start working and answering calls. To do so, the call center manager needs to activate the new hire in the system (*activate new hire*).

After 7 to 10 days of work, the new hire, the mentor of the new hire, and the call center manager meet in a *probationary review* to describe the new hire's performance. If there are problems, a *performance plan* for the new employee is created.

After investigating the call center representative onboarding business process at a high level, we now describe the three subprocesses that are part of this process:

- The hire candidate subprocess
- The activate new hire subprocess
- The facilitate new hire's orientation subprocess

## The hire candidate subprocess

Figure 2-3 shows the *hire candidate* subprocess that includes activities related to the selection of candidates and related to the candidate interview. The people involved in these activities are:

- The recruiter
- The job candidate
- The hiring manager

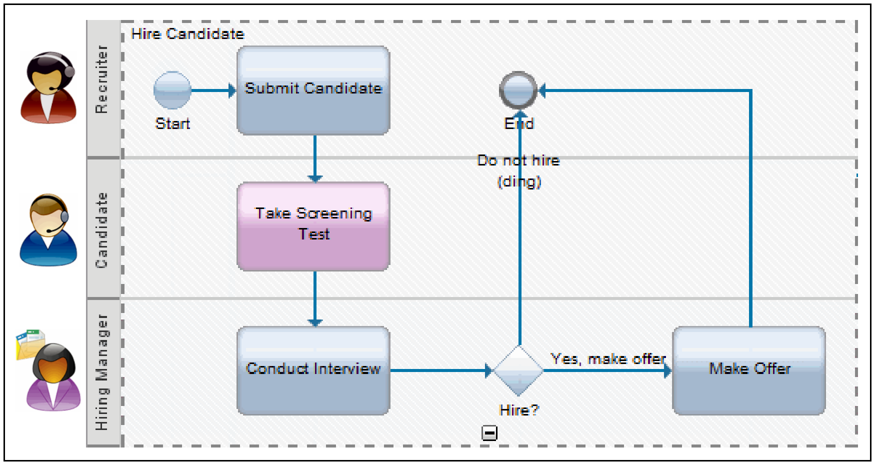


*Figure 2-3 Business scenario: Hire candidate subprocess*

The recruiter first identifies a job candidate whose profile matches the requirements of a call center representative at Call Center Company C and schedules the interview with the hiring manager and the job candidate (*submit candidate*).

The candidate first needs to accomplish a screening test, in which his language skills and typing skills are evaluated (*take screening test*).

After the test, the hiring manager performs a 30-minute job interview (*conduct interview*) with the candidate. After the interview, the hiring manager decides whether to hire the candidate based on the interview, the test score, and the candidate’s minimum asking salary.

Depending on the hiring decision, a job offer is made to the candidate (*make offer*).

The *hire candidate* subprocess ends after making an offer to the job candidate or if the decision is made not to hire the candidate.

## The facilitate new hire’s orientation subprocess

Figure 2-4 shows the *facilitate new hire’s orientation* subprocess. The process contains various activities that are all performed by the HR administrator. The HR administrator welcomes the new employees in the company and accompanies them during the first 2-3 days at the company.

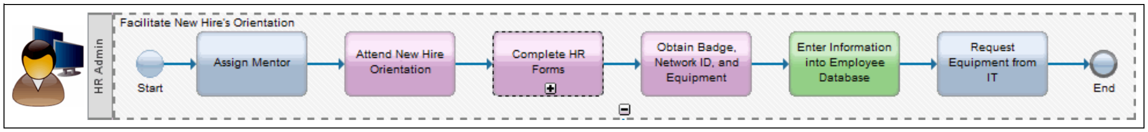

HR Admin
Facilitate New Hire's Orientation
Start
Assign Mentor
Attend New Hire Orientation
Complete HR Forms
Obtain Badge, Network ID, and Equipment
Enter Information into Employee Database
Request Equipment from IT
End

*Figure 2-4 Business scenario: Facilitate new hire’s orientation subprocess*

First, the HR administrator identifies and assigns a mentor to the new call center representative (*assign mentor*). The mentor must work in the same call center and should sit in short distance from the new employee.

After that, the HR administrator makes sure that the new hires attend the new hire orientation (*attend new hire orientation*) and complete HR forms, such as the I-9 form, tax forms, and benefits forms (*complete HR forms*). The HR forms completion is a multi-step activity. Figure 2-5 shows the details of this activity.

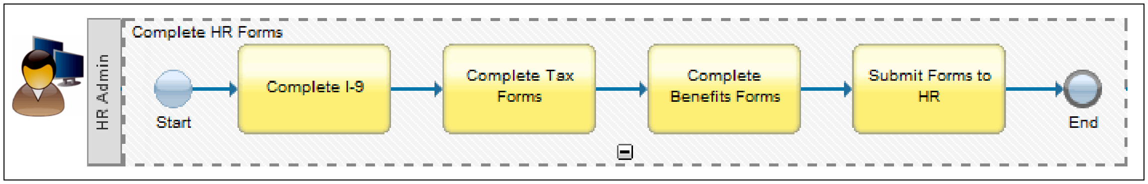


*Figure 2-5 Business scenario: Complete HR forms*

After completing and submitting the forms, the HR administrator provides a security badge, a network ID, and work equipment (for example, a headset) to the new employee (*obtain badge, network ID, and equipment*).

In the end, the employee information is entered into the employee database (*enter information into employee database*) and the IT equipment is requested (*request equipment from IT*).

### The activate new hire subprocess

The last subprocess is the *activate new hire* subprocess (Figure 2-6). It contains two activities that are performed by the call center manager:

- The first activity in the activate new hire subprocess is the *establish work schedule/shift* activity. The call center manager describes the work schedule with the new employee and inserts the new employee into the work schedule of the call center using a scheduling system. In doing so, the call center manager might shift existing schedules.
- The call center representative can start working only after the call center manager activates the profile for the employee in the call center system (*activate call center profile*). After that the new employee can accept calls.

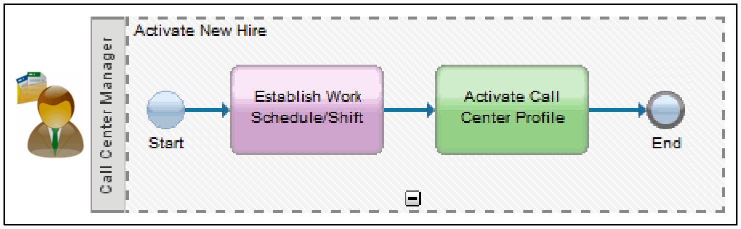


*Figure 2-6 Business scenario: Activate new hire subprocess*

## 2.4 The pain points and goals

The CEO of Call Center Company C decided to change and improve the way that call center employees are brought on board for various reasons. This section gives an overview of the pain points of the current solution and points out the goals that should be achieved with the new solution.

The pain points of the current onboarding process are:

- The onboarding of new call center representatives takes too long (around two months) because the process is manual.
- In the past, there were cases where important steps in the onboarding process were not executed. There were examples where new call center representatives could not start to work because the workplace and the equipment for them was not requested. The reason for this is that different people handle different cases, and especially in the summer when the main person in charge was on vacation, the replacements made mistakes. This situation is not surprising, as the process itself is not documented at all.

- Call Canter Company C is not able to onboard enough employees in a short time-frame and loses business because they are not able to handle the increasing amount of customer requests. The efficient onboarding of new employees is especially crucial when a large company signs a call center outsourcing deal.
- The managers of Call Center Company C need to call a number of different people to discover what the current situation is regarding the onboarding of new call center representatives. This approach is time-consuming and error-prone.

With the new onboarding solution implemented in IBM Business Process Manager V7.5, Call Center Company C wants to overcome the pain points of this situation. For the new solution, they want to achieve the following goals:

- They want to be able to onboard new call center employees (around 500) in the required time frame.
- The overall onboarding process should not take longer than 1 month. The company needs to react to changes in demand quickly.
- The costs to onboard new call center representatives should be reduced.
- The process is documented and understood by every person. It is a key success factor not to forget any steps in the process and to have clear responsibilities.
- The success rate of new call center employees who complete the probationary period successfully and do not require a performance plan needs to improve.
- It needs to be ensured that every role involved in the process has at least three people assigned to it. Less than three people would represent a high risk for a bottleneck. If one person is ill or on vacation, it cannot be allowed to have an impact on the overall duration of the onboarding process.
- The people who are involved in the onboarding of new employees (for example, the hiring manager or the facility manager) must be supported by the new system to accomplish their work. This situation includes the usage of a task list with a GUI.
- The CEO of Call Center Company C and related managers need to have better insight into the current situation with regards to the onboarding of new call center employees. They need reports that show the status and highlight critical situations.

# 3

# Process discovery

The IBM Business Process Manager (BPM) methodology advocates an iterative approach to discover, implement, deploy, and manage a business process for continuous improvement. This chapter provides an in-depth explanation of the activities (stages) of process discovery. At each transition within process discovery from process identification to the planning of a new release, there must be ample evidence and a business value proposition to justify the effort to begin the next stage in the lifecycle.

Each stage in the discovery phase of the lifecycle of a business process requires investment to gather information, assess the process, document the model, and analyze before moving onto the next stage. Figure 3-1 illustrates the four stages of process discovery and how the time and effort investment grows as you progress from one stage to the next.

The four stages of process discovery are:

1. Identify (1 - 2 hours).
   - Add process to Blueworks Live.
   - Conduct first interview with process owner.
2. Assess (2 - 3 days).
   - Conduct Process Improvement and Discovery Workshop.
   - Prepare solution proposal for stakeholder review.

3. Document (1 - 2 weeks).
   - More workshops modeling the *as-is* process.
   - Validate system integrations and IT requirements.
4. Analyze (2 - 3 weeks).
   - More workshops analyzing the *as-is* process.
   - More workshops modeling the *to-be* process.
   - Prepare and present Playback Zero.

This chapter describes the key concepts and best practices for identifying, assessing, documenting, analyzing, and maintaining the specification for a business process application.

This chapter introduces the importance of process ownership and the cultural change this concept entails and then describes the four stages of process discovery in the order in which they occur:

- 3.1, “Creating a culture of process ownership” on page 43
- 3.2, “Identifying business processes with an inventory” on page 47
- 3.3, “Assessing business processes for business impact” on page 53
- 3.4, “Discovering processes with Blueworks Live” on page 64
- 3.5, “Analyzing business processes with Blueworks Live” on page 78
- 3.6, “Next steps” on page 94

The lifecycle for a business process application begins when a process owner calls attention to their process by initiating process discovery and documentation activities. We call this first step *identification*, whereby we recognize that there is a process by naming the process and its owner in our process inventory.

Assessment begins when we consider an identified business process for further investigation. The key impetus in the decision to continue investing in discovery activities is an assessment of the business impact and how the impact compares to the impact of other processes.

Although documenting a business process begins as soon as you identify the process with a name and process owner, there is a stage during discovery that involves interviews with participants and stakeholders. These interviews allow you to understand and document the process model, user stories, key performance indicators, problems, risks, pain points, and critical success factors.

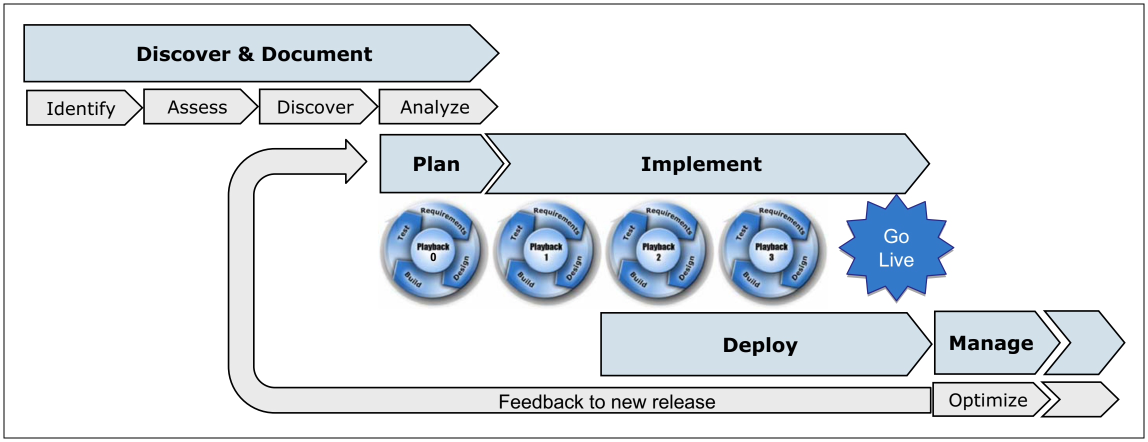


*Figure 3-1 Identifying a process leading to a solution approach initiating planning and implementation*

Analysis begins only after fully understanding the boundaries, goals, and problems of the as-is process. Process analysis leads to a solution approach and a proposed to-be process.

In an enterprise-wide BPM program where process owners continuously identify, assess, discover, and analyze dozens of business process applications, the outcome of each stage helps prioritize candidate processes for planning and implementation. The outcomes of process discovery and documentation include the business case and Playback Zero that help prioritize each business process application against others for planning and implementation. Process analysis activities often continue during planning and early implementation and continue through Playback One.

## 3.1 Creating a culture of process ownership

Process ownership is essential in BPM. Your BPM projects will not be successful without a process owner, and your BPM program will not succeed without creating a culture of process ownership. This cultural transformation requires a paradigm shift in traditional IT thinking whereby technologists (IT, vendors, partners, and so on) provide customers (operational business) with applications and services as requested, and paid for, by the operational business.

In BPM, the conversation shifts away from delivering products and services to the customer to a conversation about business value. This value is realized by the operational business, not in the delivery of tools and services, but in the success of a *business process*. Technologists still participate in and facilitate the delivery tools and services to support the discovery, planning, implementation, deployment, management, and governance of a business process, but the conversation, at every juncture, remains focused on business value.

**Paradigm shift:** The paradigm shift that happens in a BPM transformation changes the dynamic between operational business and IT from a customer/vendor relationship to a true partnership in which all parties remain focused on the business value realized with a *managed process* that supports corporate objectives.

### 3.1.1 Process ownership must become a cultural phenomenon

For many organizations, the concept of *process ownership* is new. Managers own people, they own systems and applications, they control budgets, and they are accountable to the business unit for quarterly targets. Yet it is often difficult for a manager to name the *processes* that they own. It is difficult enough to recognize and name processes that are internal to business units or departments, much less name the owners of those processes that cross departmental lines.

During interviews with process participants and discussions concerning the scope, risk, and pain assessments, the process owner often becomes self-evident. The process owner is the person most interested in the success or the outcome of the process. This person has the most to gain (or lose) when the process is a success (or a failure).

You can transform your business culture to recognize business processes and their owners by staying focused on business value. In our business scenario, Call Center Company C learns to recognize that hiring, training, and retaining call center representatives is a primary activity on their enterprise value chain. Focused on the output of this process (trained call center reps) the hiring manager quickly recognizes that the process by which the corporation *produces skilled resources* belongs to the manager. The manager alone stands to gain the most if that manager can manage this process, reduce lead times, reduce cost per new hire, and increase the quality of the new hires as measured by performance and retention.

## 3.1.2 Process ownership changes a manager's perspective

In the scenario in this book, the hiring manager is the process owner of the *call center rep onboarding* process (Figure 3-2).

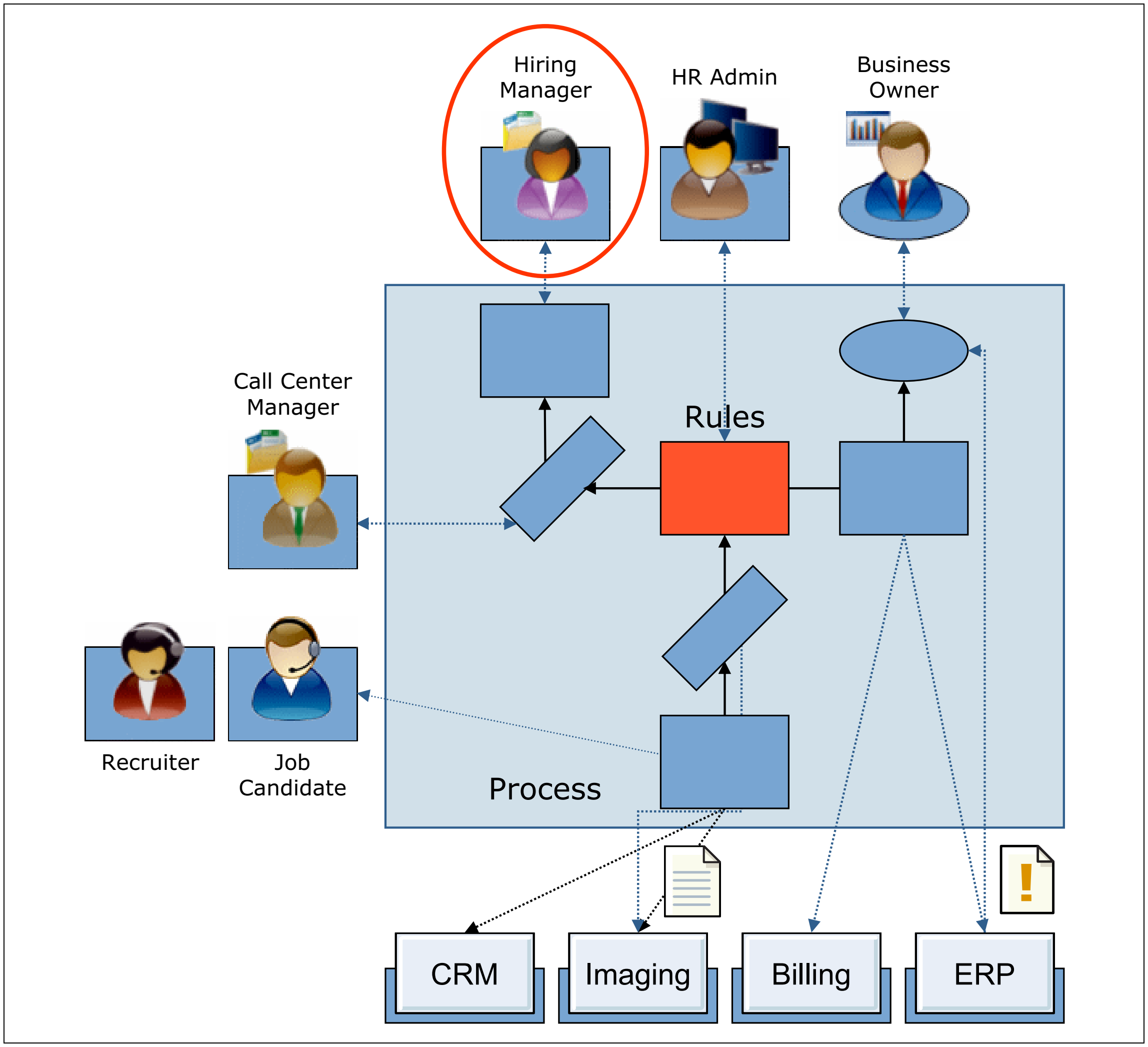


*Figure 3-2 The hiring manager is the process owner of call center rep onboarding process*

Accountable to the business owner (VP call center operations), the hiring manager is responsible for reducing lead times in producing new hires, improving quality of new hires as measured by performance and retention, and reducing the cost to recruit, hire, train, and retain new hires.

Before introducing BPM, the hiring manager felt responsible for the following resources and activities. The manager did not view the manager role as being a key contributor to corporate objectives.

- IT systems used to track recruits and new hires
- New hire training courses and training materials
- Scheduling new hires for required training courses
- Greeting new hires on their first day
- Maintaining data integrity on new hire information
- Submitting feature requests to IT
- Working with contract recruiters
- Monitoring contract recruiters subjectively based on communication skills, response time to job requisitions, and number of candidates submitted

With BPM, the hiring manager’s perspective is transformed. The manager has a key role in a primary activity on the value chain that directly supports corporate strategy to grow the business, sign larger clients, and increase margin on outsourced call center services by hiring, training, and retaining high-performing call center representatives. The call center rep onboarding process is the manager’s responsibility and is objectively measured by the output of that process: high-performing call center reps.

With a BPM perspective, she also has an eye on key performance indicators that monitor her process and correlate her success with core business objectives. With this new perspective, certain elements that used to be important, such as feature requests for enhancements to the HR system, became far less important than *managing* and *optimizing* the process used to hire, train, and retain high-performing call center representatives.

As shown in Figure 3-2 on page 45, the hiring manager has a new perspective of the business process and its importance on the value chain. The hiring manager takes ownership of the business process and its outcomes at a higher level than simply taking responsibility for the people and systems involved in the process. This new perspective gives the hiring manager a new sense of accountability to the company’s core business and new inspiration to succeed.

## 3.2 Identifying business processes with an inventory

Maintaining an inventory of business processes improves the likelihood of meeting business objectives. With an inventory, you can plan and trace your process discovery activities to core business objectives. This situation improves your visibility and helps determine what processes to work on and when. You want to work on the correct processes at the correct time.

### Minding your business: Knowing your processes

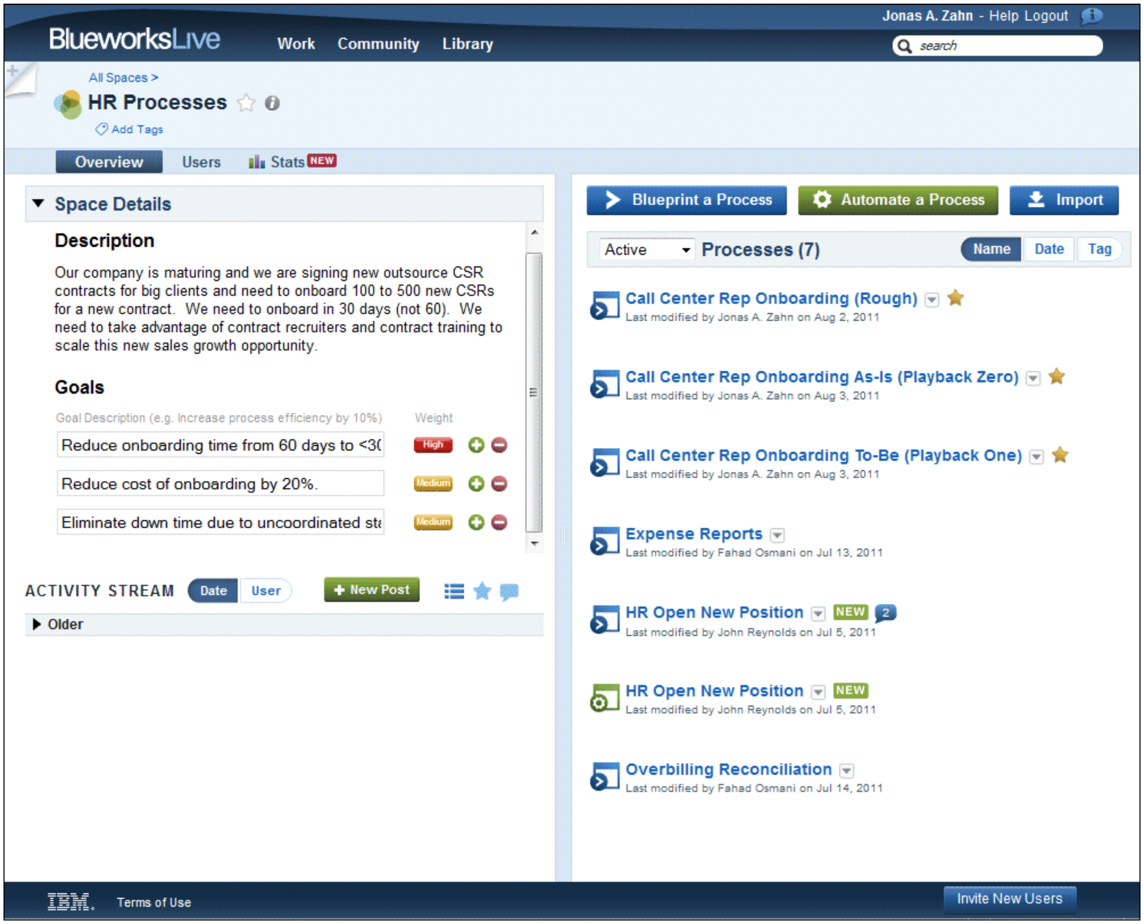
A process inventory exercise quickly generates a list of business processes (or process areas) that can be categorized by business unit, size, business impact, risk, and pain. In a brief interview, you gather enough details for each process to map it to the value chain and help prioritize processes for further assessment, discovery, and analysis. Details include the process owner, a 3 - 5 sentence description of the process, a rough estimate of size and complexity, the risk associated with the process, and the level of pain experienced in the as-is process.

Process details needed for identification and prioritization for further assessment and discovery are:

- Process owner
- Short description (3 - 5 sentences)
- Size and complexity
- Risk
- Pain

The value chain, described later in this section, is a good generic map that helps identify processes and where they fit in the overall landscape of processes in your enterprise. There are, however, dozens of industry-specific categorical hierarchies of business processes. One example is the insurance industry standard ACORD (http://www.acord.org/) capability model that contains a list of named processes common to insurance organizations. There are similar models in manufacturing, financial services, and many other industries. Using one of these standard models might help accelerate your process inventory.

Figure 3-3 on page 48 shows where to start documenting processes with a process inventory.



**Description**

Our company is maturing and we are signing new outsource CSR contracts for big clients and need to onboard 100 to 500 new CSRs for a new contract. We need to onboard in 30 days (not 60). We need to take advantage of contract recruiters and contract training to scale this new sales growth opportunity.

**Goals**

| Goal Description (e.g. Increase process efficiency by 10%) | Weight |
|---|---|
| Reduce onboarding time from 60 days to <3( | High |
| Reduce cost of onboarding by 20%. | Medium |
| Eliminate down time due to uncoordinated st: | Medium |

- Call Center Rep Onboarding (Rough) — Last modified by Jonas A. Zahn on Aug 2, 2011
- Call Center Rep Onboarding As-Is (Playback Zero) — Last modified by Jonas A. Zahn on Aug 3, 2011
- Call Center Rep Onboarding To-Be (Playback One) — Last modified by Jonas A. Zahn on Aug 3, 2011
- Expense Reports — Last modified by Fahad Osmani on Jul 13, 2011
- HR Open New Position NEW 2 — Last modified by John Reynolds on Jul 5, 2011
- HR Open New Position NEW — Last modified by John Reynolds on Jul 5, 2011
- Overbilling Reconciliation — Last modified by Fahad Osmani on Jul 14, 2011

*Figure 3-3 Start documenting processes with a process inventory*

## 3.2.1 Maintaining a process inventory with Blueworks Live

Blueworks Live is an enterprise-class process modeling tool. Available as a service (SAAS), there is no software to download or install to get started with Blueworks Live. You can invite collaborators to review, edit, and comment on processes by email. Blueworks Live is the fastest way to start a process inventory and keep it current through collaborative participation across your enterprise.

Blueworks Live can be found at the following address:

`https://blueworkslive.com/`

## Why maintain a process inventory

In the modern sense, the term *inventory* represents the goods and materials held by a business, often made available for sale, and invokes imagery of real-time accounting and accuracy for each unit of stock held by the business. In the classic sense, an *inventory* is a list compiled for a specific purpose. A common usage of the word refers to the list of furnishings in a home to be rented. This list does not need to contain every last furnishing, but it does contain those furnishings that must be repaired or replaced if they are damaged by the tenant.

It is in this classic sense that you build an inventory of business processes in your organization. Do not concern yourself with adding *every* business process to your inventory. You need to list only those business processes that are relevant to your business objectives today. The tools that you use to keep a process inventory should make it easy to organize and find new and common processes, to share and collaborate on processes, and to control who has access to view or change processes.

Key reasons for maintaining a process inventory are to:

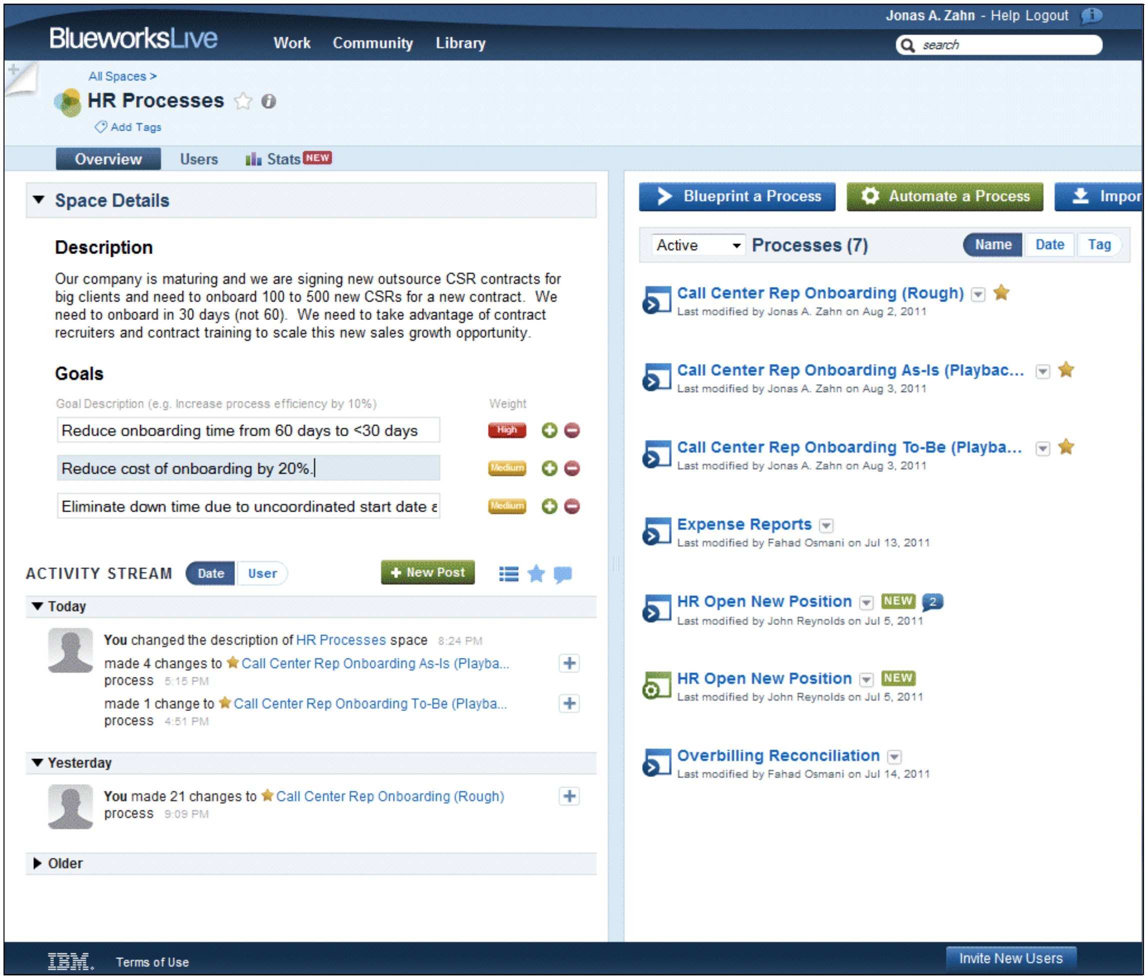
- Organize and easily find business processes.
- Share and collaborate on business processes.
- Control access to viewing or changing business processes.

## Getting started with Blueworks Live

Blueworks Live makes it easy to start maintaining a process inventory with contributions from hundreds of collaborators throughout your organization. The collaborative modeling and analysis tools are available as a service with no software to download or install. In minutes, you create an account and invite process owners and process participants to collaborate.

The first thing to do in Blueworks Live is create a *space* in which you define business processes. A space could represent a narrow process area (that is, recruiting or new hire training), a business unit (that is, new hire intake), or an entire department of our business (that is, human resources). Once created, add a few sentences to describe the new space and create a few goals that characterize the reasons for wanting to document the processes for this part of the business.

Figure 3-4 shows a new space in Blueworks Live for human resource processes.



Space Details

Description

Our company is maturing and we are signing new outsource CSR contracts for big clients and need to onboard 100 to 500 new CSRs for a new contract. We need to onboard in 30 days (not 60). We need to take advantage of contract recruiters and contract training to scale this new sales growth opportunity.

Goals

| Goal Description (e.g. Increase process efficiency by 10%) | Weight |
|---|---|
| Reduce onboarding time from 60 days to <30 days | High |
| Reduce cost of onboarding by 20%. | Medium |
| Eliminate down time due to uncoordinated start date ε | Medium |

Today

You changed the description of HR Processes space 8:24 PM
made 4 changes to ★ Call Center Rep Onboarding As-Is (Playba... process 5:15 PM
made 1 change to ★ Call Center Rep Onboarding To-Be (Playba... process 4:51 PM

Yesterday

You made 21 changes to ★ Call Center Rep Onboarding (Rough) process 9:09 PM

Older

Call Center Rep Onboarding (Rough) ★
Last modified by Jonas A. Zahn on Aug 2, 2011

Call Center Rep Onboarding As-Is (Playbac... ★
Last modified by Jonas A. Zahn on Aug 3, 2011

Call Center Rep Onboarding To-Be (Playba... ★
Last modified by Jonas A. Zahn on Aug 3, 2011

Expense Reports
Last modified by Fahad Osmani on Jul 13, 2011

HR Open New Position NEW 2
Last modified by John Reynolds on Jul 5, 2011

HR Open New Position NEW
Last modified by John Reynolds on Jul 5, 2011

Overbilling Reconciliation
Last modified by Fahad Osmani on Jul 14, 2011

*Figure 3-4 New space in Blueworks Live for human resource processes*

Although there is no inherent limit on the number of business processes that can be defined in a single space, it is helpful to understand that a space contains users with permissions to blueprint or automate all processes in that space. We suggest creating spaces organized around the groups of business people who created the blueprint for the business processes in that space.

As your organization transforms and begins to adopt a culture of process ownership, you might find yourself in the company of dozens or hundreds of collaborators in Blueworks Live adding new processes and making updates to those processes daily. Blueworks Live makes it easy to find new processes, identify common processes, and reconcile overlapping processes with tools that provide visibility with spaces, tagging, and archiving processes.

Blueworks Live also makes it easy for process owners to recognize and reuse common processes. One key to scaling BPM adoption lies in enabling cross-departmental and cross-functional visibility and collaboration with tools like Blueworks Live and IBM Business Process Manager V7.5. When teams of people from different areas of the business collaborate and reuse process assets, their ability to plan, implement, and deploy new business processes improves. Process owners realize shorter delivery cycles with reuse, faster time-to-value with cross-departmental collaboration, and improved return on investment (ROI) with alignment to corporate strategy. All of these items are possible through the new level of visibility made possible by BPM, a culture of process ownership, and the tools to manage business process.

## 3.2.2 Conducting the first interview

A brief interview, conducted in one hour or less, with process owner and process participants (that is, subject matter experts (SMEs)) provides the details needed to identify the process and add to your process inventory. This first interview needs to capture only a minimal description of the process without steps. Imagine a single box for the named process with start and end events (Figure 3-5). An inventory should characterize the scope, pain, and risk of the process area to prioritize against the value chain and help you select those processes to focus on and proceed with a more detailed process discovery. During a brief interview, add the new process to Blueworks Live, identify the process owner and SMEs, and describe where the process fits into the value chain.

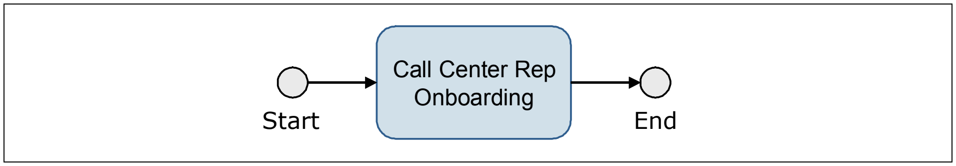


*Figure 3-5 A single box to name and describe the process and identify the owner*

## 3.2.3 Setting a compass bearing with value chain analysis

First described and popularized by Michael Porter in his 1985 best-seller *Competitive Advantage: Creating and Sustaining Superior Performance*, the *value chain* is a business management concept that generically categorizes value-adding activities of an organization. This approach to process discovery helps identify key processes (activities) and where and how those processes impact the core business (Figure 3-6).

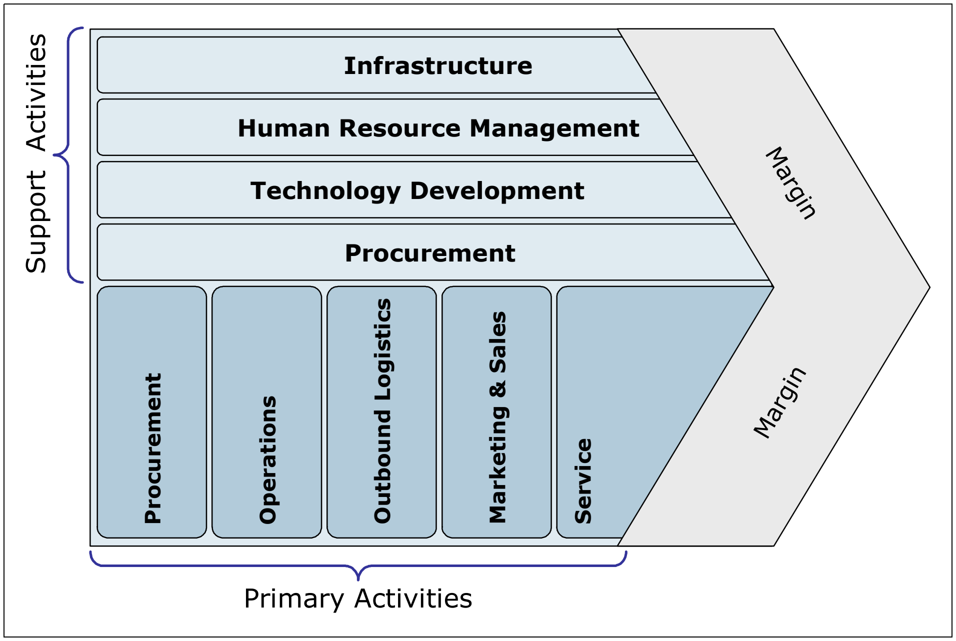


*Figure 3-6 Use value chain analysis to position a business process*

The value chain illustrates the importance of recognizing the difference between primary activities and support activities. The primary activities of an enterprise include inbound logistics, operations and production, outbound logistics, marketing and sales, and services (maintenance). Processes for managing human resources, administration, infrastructure, technology, and procurement are support activities.

This situation does not mean that your BPM projects should target only primary activities. There could easily be support processes in need of automation, visibility, and management that are good candidates for BPM. If a support process has a business case and an ROI in alignment with corporate objectives, it is a valid candidate for a BPM project.

Use value chain analysis to orient your business processes on the landscape for your enterprise. With the value chain, you create a common reference and foundation for building the business case to justify planning and implementation for your business process.

## 3.3 Assessing business processes for business impact

For your first BPM projects, you decide to proceed with discovery and analysis work by selecting the process. But after your first process implementations and as your culture begins to recognize business processes and process owners, you will have no shortage of candidate processes for implementation. Which process do you work on next? You need to prioritize your business processes and select those processes for the initial investment of discovery and documentation. Having a decision gateway in process selection for further discovery and analysis is an indication of organizational maturity on your BPM journey.

### 3.3.1 Conducting a process improvement and discovery workshop

The Process Improvement and Discovery Workshop (PIDW) is an assessment tool that, in less than three days of time invested, can help determine whether the process has a valid business case and an ROI worth pursuing further. The PIDW aims to start preliminary discovery and document the initial business impact and determine technical feasibility to implement and deploy a process solution. This workshop sets an accurate level of expectation for both the delivery participants (IT) and operational business participants (business process owner and SMEs). The objectives of this 2 - 3 day workshop activity are described in the following sections

#### Identifying

Through collaborative discussion and documentation in Blueworks Live, the process and process owner are again verified. Workshop participants identify and document the business drivers and begin to form a perspective that places this process on the enterprise value chain and traces to corporate objectives. The business process success factors should be clearly stated in the process description. Process risks and issues should characterize the business impact if the process remains As-Is.

### Understanding

Both business participants (process owners and SMEs) and technology participants (delivery team and architects) must have a common understanding of the business process. This means that all participants share a common understanding as to the process definition, the business impact, and how to measure that impact. Participants also gain a common understanding of the size, scope, and complexity of that process, and a vision for how to apply IBM Business Process Manager V7.5 to solve the business problem and deliver measured business value.

### Assessing

Assessing a business process begins by mapping it on the enterprise value chain and aligning to corporate initiatives. Teams often gain a broader perspective on the impact and importance of the process as they begin to outline the business case. Process assessment validates key performance indicators (KPIs) and service level agreements (SLAs) and how they might roll up to the enterprise to support corporate objectives. For many teams, this assessment exercise uncovers large value statements, further justifying the implementation of the business process.

> **Business impact:** Be careful not to limit assessment of business impact to the value gained through *process automation* alone. With a limited perspective, value statements often pertain only to task efficiency, team productivity, and business throughput. There is often additional and significant business impact hidden in new management capabilities afforded by *visibility* and *control* that might lead to improved customer satisfaction, higher quality output, and more revenue opportunity.

### Defining

The workshop participants collaborate to define the business process, proposed approach, and next steps for a successful project. Discovery work includes proposing a solution approach and solution design to deliver business value. The output of this workshop should also include next steps to plan and implement a successful BPM project.

### Preliminary analysis

During the 2 - 3 day workshop, there is little time to for analysis and process improvement work, but opportunities should be identified and documented for further discovery and analysis. There might be simple changes that can be made to the As-Is process to realize some improvement. These changes can be proposed, played back to the workshop participants for validation, and documented. It is important, however, to set expectations that a 2 - 3 day workshop does not produce an optimized process.

Process improvement is a long journey and should not stop with PIDW. A PIDW by itself is not a process improvement exercise. Process improvement does not materialize until the process is managed. There is little incentive to follow or improve the process if not managed. In terms of implementation, this situation means that when we *deploy* an executable business process and start to monitor it, only then can we begin to manage and optimize the process.

Only after deploying a process can we begin to manage and optimize the process (Figure 3-7).

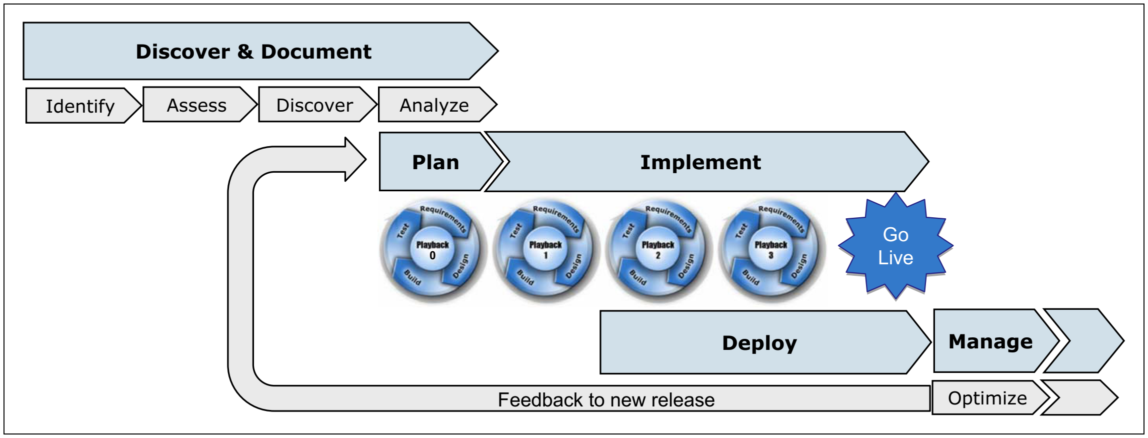
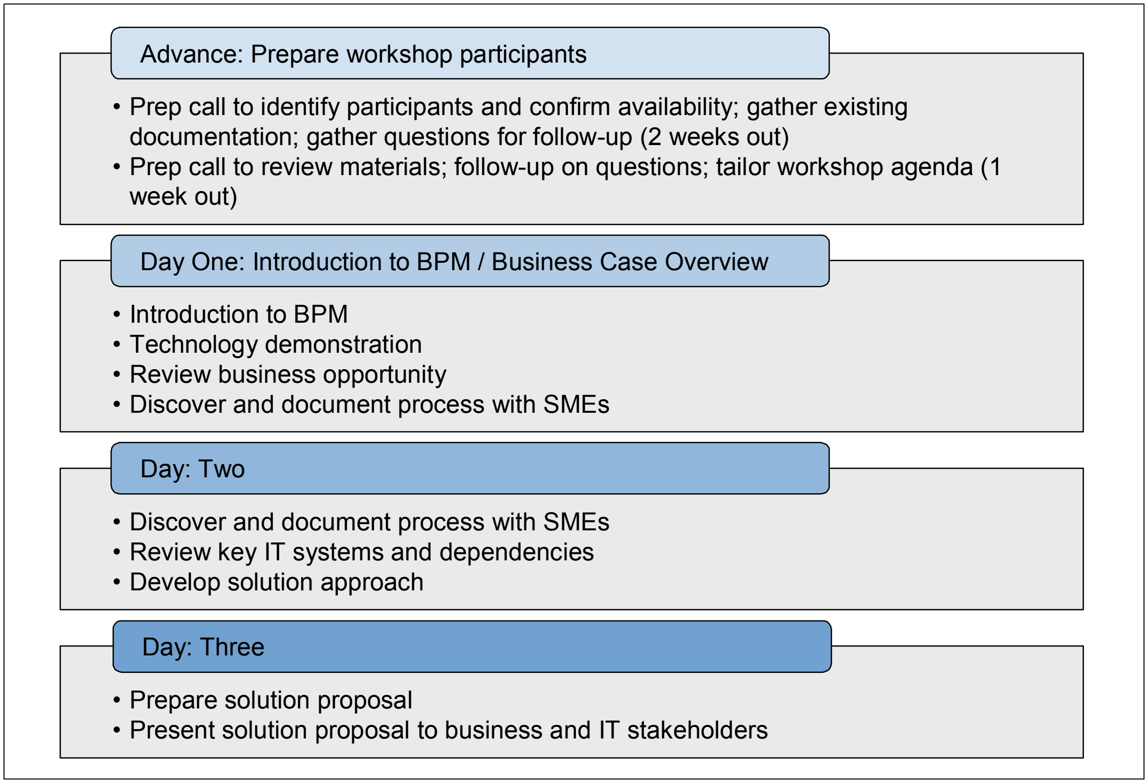


*Figure 3-7 Only after deploying a process can we begin to manage and optimize the process*

## 3.3.2 Agenda for a Process Improvement and Discovery Workshop

The following is a sample agenda of activities for preparing and conducting a PIDW (Figure 3-8):

1. Prepare workshop participants.
2. Introduce BPM.
3. Identify: Business case overview.
4. Assess: Determine business impact.
5. Define: Document the business process.
6. Develop the solution approach.
7. Discuss findings and proposed solution.



Advance: Prepare workshop participants

- Prep call to identify participants and confirm availability; gather existing documentation; gather questions for follow-up (2 weeks out)
- Prep call to review materials; follow-up on questions; tailor workshop agenda (1 week out)

Day One: Introduction to BPM / Business Case Overview

- Introduction to BPM
- Technology demonstration
- Review business opportunity
- Discover and document process with SMEs

Day: Two

- Discover and document process with SMEs
- Review key IT systems and dependencies
- Develop solution approach

Day: Three

- Prepare solution proposal
- Present solution proposal to business and IT stakeholders

*Figure 3-8 Example of discovery workshop agenda*

### Preparing workshop participants

Schedule a prep call or meeting at least two weeks in advance to verify availability of all required participants. Review the business context, existing documentation, and process models. Gather questions and assign to participants to gather additional information.

Meet again one week in advance to review open questions, give additional information-gathering assignments, and finalize the workshop agenda. If additional data is required, or if one or more participants are not available, reschedule the workshop.

**Scheduling a PIDW:** Never schedule a PIDW without the process owner and SMEs. Without an empowered decision maker and the experts in the workshop, the output of the workshop is less than useful.

## Introducing BPM

For workshop participants who are new to BPM, provide an overview of BPM and the tools of IBM Business Process Manager with a demonstration in a 1 - 2 hour discussion and presentation. Kick off the workshop with a review of the business opportunity and outline the business value case for this process area.

## Discovering and documenting the process

Engage participants in a hands-on collaborative workshop in Blueworks Live. The workshop leader provides basic guidelines of good process modeling. This activity occupies 1 - 2 days, or most of the workshop time, in 3 - 4 hour sessions with the process owner and SMEs.

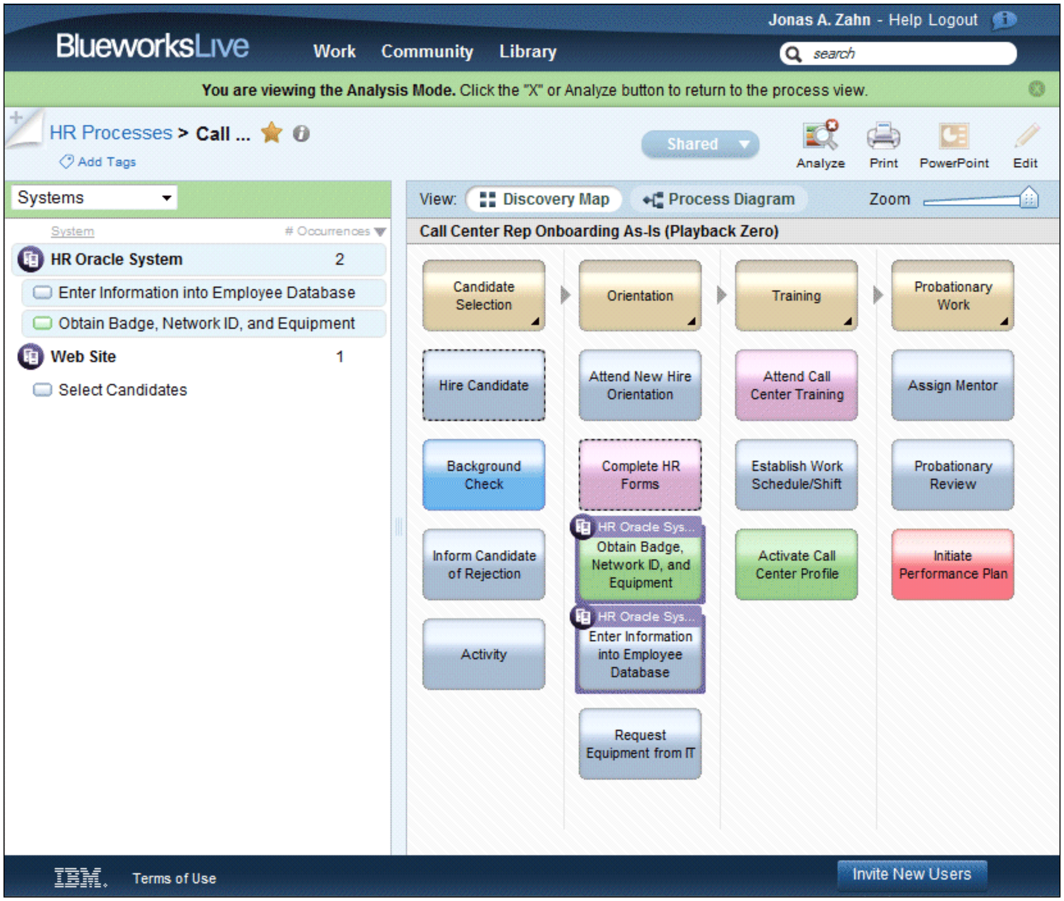
What are some of the first principles to aim to use in your first project?

- Model for discovery first, but also keep the model “theoretically implementable” as you go. This situation means allowing participants to brainstorm and tell their stories and capture them as told, but refrain from documenting proposed “I want” activities that you know are not feasible.
- Document key performance indicators (KPIs), success metrics, and pain points for the project. These items start small but should quickly become relevant to your enterprise value chain.
- Use the Suppliers, Inputs, Process, Outputs, Customers (SIPOC) method for defining the business details for each activity in the process. Blueworks Live incorporates SIPOC, so make full use of Blueworks Live and start building a process inventory.

## Determining scope and complexity

Perform a rough count of steps (activities and subprocesses), participants (swim lanes), user interfaces (coaches), system integrations, and business entities (complex variables) to determine the overall complexity of the process. With a fairly complete SIPOC analysis and activity details completed in Blueworks Live, it is easy to determine the complexity of a business process.

Use the *analyze* tool in Blueworks Live to estimate the complexity of the process as it involves different participants, systems, process owners, experts, suppliers, customers, inputs, and outputs. Figure 3-9 shows the usage of analyze mode, which reveals that there are a total of three system references and two distinct systems with which our process must integrate. At this time, we might have not determined *how* to integrate to these systems, just that the process depends on saving or retrieving information to and from these systems.



**You are viewing the Analysis Mode.** Click the "X" or Analyze button to return to the process view.

| System | # Occurrences |
|---|---|
| HR Oracle System | 2 |
| Enter Information into Employee Database | |
| Obtain Badge, Network ID, and Equipment | |
| Web Site | 1 |
| Select Candidates | |

*Figure 3-9 Count of system integrations in analyze mode in Blueworks Live*

## Developing the solution approach

A BPM solution architect works with the team to prepare a high-level solution approach to the business problem. This solution approach includes a rough order estimate of the number of different business process definitions (BPDs) and the relative number of steps (subprocesses and tasks) in the solution.

The solution architect uses the rough count of the number of system integrations required and the opportunity for trade-offs with system integrations (that is, swivel-chair integrations). As you develop a solution approach, you begin to outline the feasibility and cost/benefit analysis of each system integration. While some integration decisions might be easy, others might not be decided until implementation begins. In either case, the solution approach should create a shared vision as to how the process might be implemented and a rough-order-magnitude (ROM) estimate for what the implementation might require.

### Discussing findings and proposed approach

In the final 3 - 4 hours of the workshop, the process owner, with help from the solution architect, presents the workshop summary to stakeholders. It is important that this presentation is a joint presentation mainly from the process owner (versus *to* the process owner), as this presentation sets the collaborative tone for the project and highlights the importance of *active* participation from the process owner.

## 3.3.3 Pain assessment

Process owners often bring their business processes forward for consideration due to some type of pain or complaint. As we add processes to our process inventory, we describe and document that pain.

The process owner experiences many types of pain today, including:

- Lack of visibility

  For example:

  - The process owner does not know what is happening in the process.
  - The process owner cannot see problems until it is too late.

- Lack of management

  For example:

  - The process owner has difficulty getting status reports.
  - The status reports do not really help the process owner take corrective action.

- Lack of efficiency

  For example:

  - It takes too long to do the work due to complex systems.
  - It takes months to train and years to become a proficient user.
  - Lots of rework. The same data is entered multiple times in multiple systems.

- Lack of coordination

  For example:

  - The “left hand” does not know what the “right hand” is doing.
  - Two different departments call the same customer with conflicting information.

- Lack of consistency

  For example:

  - Different users do the same thing different ways.
  - Quality of output is inconsistent.

## 3.3.4 Risk assessment

There are many business books written on risk assessment methods, and this book makes no attempt to create or endorse a method of quantifying and documenting risk. Our message is about the importance of using any method or some means to consistently assess the risk of the business process candidate. What is the level of risk in the process area? Does the pain in the as-is process expose the enterprise to regulatory/legal risk, market exposure, or competitive risk? How does the risk trend? Does this risk continue to get worse, get better, or stay the same over time? Assessing the risk of the process area helps prioritize the process for further discovery and analysis.

In addition to documenting the *business* risks within the process (or process area), you should also characterize the *delivery* risks associated with implementing the business process or sensitivity to corporate politics. Carefully consider pursuing a project if any of the following conditions exist that would characterize the implementation for this process as high risk.

The types of processes that might be high risk to success with BPM adoption are:

- A single process owner is not identified or not available. For example:
  - We are having trouble resolving issues by committee.
  - Decisions are made only to be undone by protest.
- A process is undiscovered. For example:
  - We knew so little when we started, and now the process is far too complex to continue. We are stuck.
  - We did not fully understand the business case before we started.
- There is no need or desire to make it better. Automate the process. For example:
  - We automated a poor process and now we do more wrong faster.
  - Where is the business impact? What can we measure to improve?
- A simple process that is a proof-point for BPM. For example:
  - We implemented a simple process, but there is limited visibility and no real business impact. We are not getting noticed.
  - We proved with BPM that we can measure and reduce the time that it takes to select an employee of the month, but we have no real ROI to show for our effort.

### 3.3.5 Process selection for discovery and analysis

With an inventory of business processes, each documented to include the process owner, experts, description, pain, and risk assessments, you can select the processes and proceed with a discovery workshop. You create the business value proposition that justifies the continued effort to plan and implement a solution for the business process. In the beginning of your BPM transformational journey, select those business processes that have a high ROI. These processes have limited scope and complexity and might also have an highly measurable impact on business operations. Select processes that you know more about, have more experience with, and have ready access to the process owner and SMEs to help discover and document the process.

In the beginning, the characteristics of a candidate process might be obvious, and it is easy to justify continued effort on the business process. As we mature on our BPM transformational journey, we learn how to better recognize a business process in our operations (versus a system, an application, or a single-user task/workflow). As our business process inventory grows, the selection criteria that we use to differentiate our processes need more rigor and objectivity. The set of selection criteria described in the following sections is easy to assess at the beginning of our journey and suitable for value-added measures as we improve our own skills at measuring and quantifying business results on our BPM transformational journey.

### Business impact: Where the business value is

Your first objective is to answer the question of how you quantify and measure the impact to the business. Can the business impact be measured in head count efficiency? Will the business impact be realized in customer satisfaction? Might the process impact cost recovery or reduced expenses? Can the business impact be measured by increased revenues? After you understand how you measure the business impact, you can calculate or estimate the expected impact. What is the opportunity to change or improve the business? How might a BPM solution address the pain points?

*Table 3-1 Ways to measure business impact at Call Center Company C*

| Pain assessment | How to measure | Business impact |
|---|---|---|
| High turnover in human resources due to poor job satisfaction. | Employee turnover is up XX%.<br>Exit interviews show number one reason for leaving is job satisfaction. | Turnover leads to more hiring and training, which costs $XXX per new hire. |
| Customer satisfaction at an all-time low. | XX% of customers and YY% of sales dollars are dissatisfied.<br>Sales growth is ZZ% below forecast. | Customer referral business is at an all-time low. Repeat business is down. Sales growth is down XX%. |
| Onboarding new clients takes more than 60 days. | Days after executed agreement to full-service billing for clients requiring more than 50 call center reps is > 60 days. | Lost revenue due to inability to hire and train resources. |

## Scope and complexity: Delivering business value in 90 days

Can there be a successful implementation for a first release in 10 - 16 weeks? Some *green field* business processes have no prior software implementations to replace, migrate, or integrate. The As-Is process might be an inefficient, invisible, unmanaged business process with process variation and rework ready for a quick-win release. This type of green field application can deliver immediate value by eliminating process variation. More importantly, simple process instrumentation and tracking immediately provide the visibility necessary to begin managing the process. A process like this one might be much more attractive than one plagued by the politics from previous attempts to fix or automate the process, or one that has hundreds of steps networked in a series of processes and subprocesses involving a long list of different business data systems of record.

## Appetite for BPM: Are we ready to work

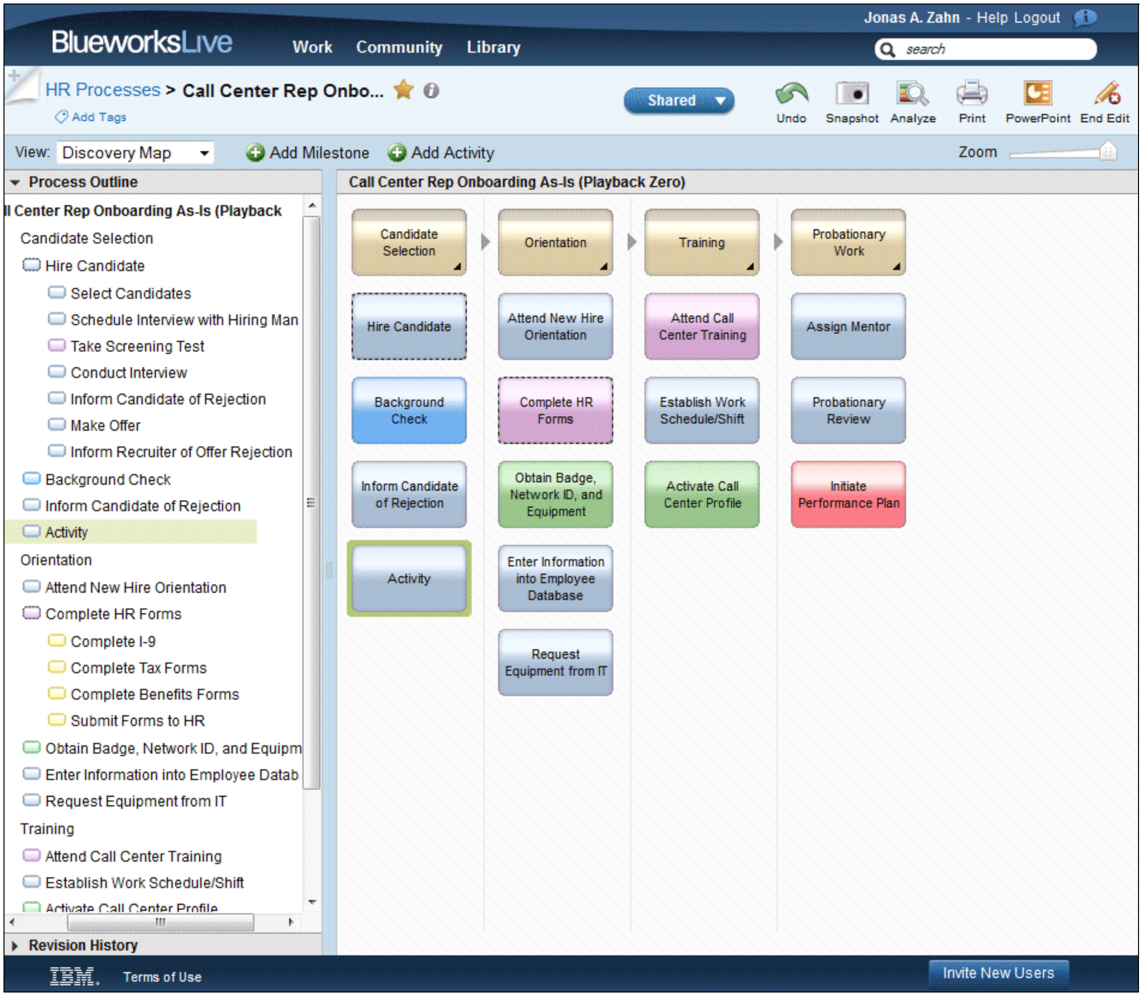
The process owner must be committed to the methodology and to the success of the implementation, and be available to participate in discovery, design, and development activities. This responsibility requires 80 -100% time commitment for the first 4 - 8 weeks of the project, including the discovery and planning phases, and roughly a 50% time commitment for the duration of the project. This responsibility can be delegated to a subordinate, but that delegate must have the full authority to decide about scope and functionality or the cadence of development is slowed by indecision.

**Important:** Process discovery, planning, and implementation without a documented and committed process owner is sure to fail.

The SMEs, often star participants in the process, must be committed to the methodology and the success of the implementation. The time requirement for these experts during discovery can vary from 20 - 100% depending on the size of the overall process and the number of expert participants. In this methodology, SMEs work alongside process analysts and process developers to discover, document, and design the solution. The daily participation of SMEs is essential to the success of the project. A process that involves dozens of different participant groups, each with globally distributed SMEs, is much more difficult to discover, plan, and implement than a process with two to three SMEs collocated with the delivery team.

## 3.4 Discovering processes with Blueworks Live

Like brainstorming, there are ground rules to process discovery. All participants in a process discovery workshop are free to express their experiences by telling their stories. Participants should be encouraged to do so regularly, freely, and without judgment. Process discovery is a collaborative activity whereby all members should be logged on to Blueworks Live (https://blueworkslive.com) and actively participating in the creation and definition of milestones and activities in the process model (Figure 3-10). As with brainstorming, this period is not the time to judge, design, or optimize the process.



ll Center Rep Onboarding As-Is (Playback

- Candidate Selection
  - Hire Candidate
    - Select Candidates
    - Schedule Interview with Hiring Man
    - Take Screening Test
    - Conduct Interview
    - Inform Candidate of Rejection
    - Make Offer
    - Inform Recruiter of Offer Rejection
  - Background Check
  - Inform Candidate of Rejection
  - Activity
- Orientation
  - Attend New Hire Orientation
  - Complete HR Forms
    - Complete I-9
    - Complete Tax Forms
    - Complete Benefits Forms
    - Submit Forms to HR
  - Obtain Badge, Network ID, and Equipm
  - Enter Information into Employee Datab
  - Request Equipment from IT
- Training
  - Attend Call Center Training
  - Establish Work Schedule/Shift
  - Activate Call Center Profile

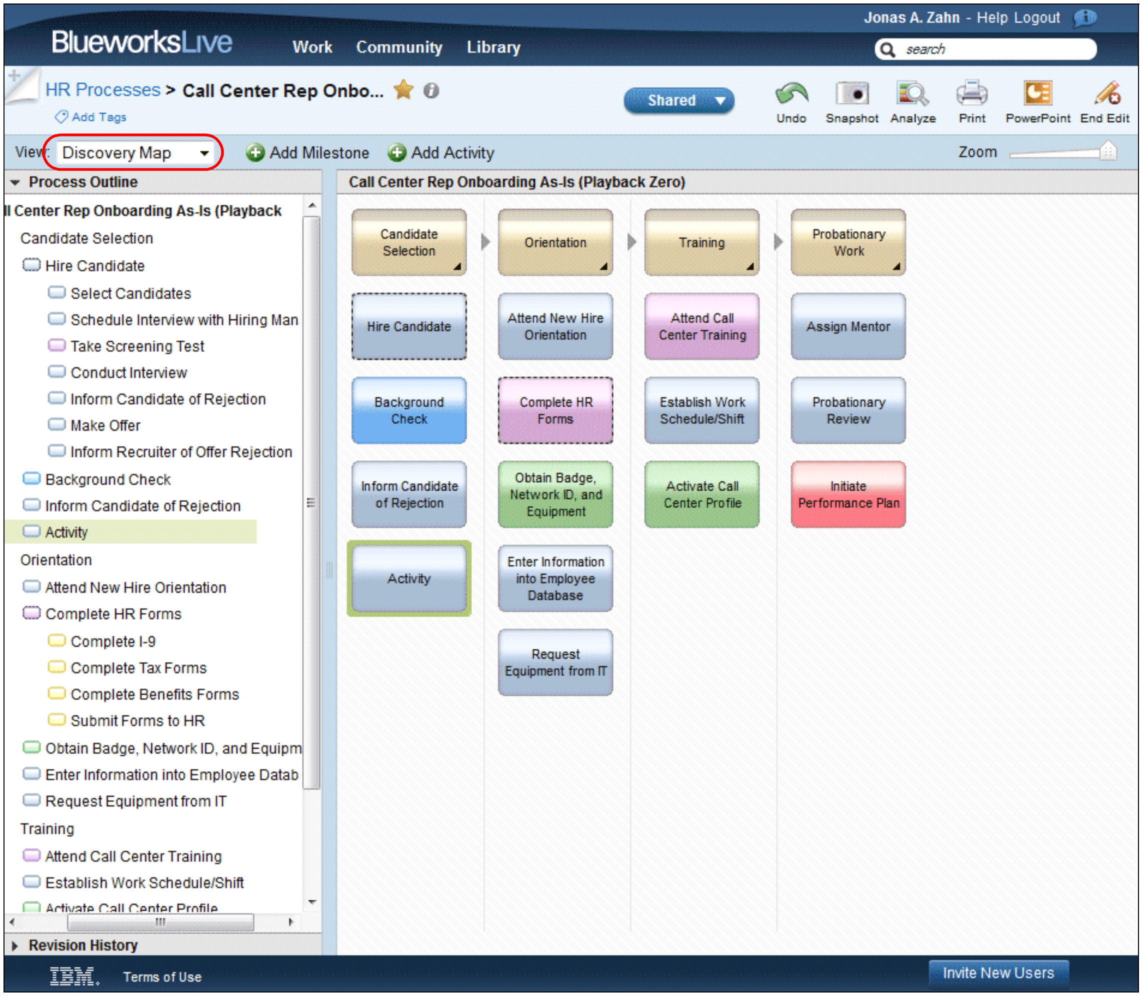
Call Center Rep Onboarding As-Is (Playback Zero)

| Candidate Selection | Orientation | Training | Probationary Work |
|---|---|---|---|
| Hire Candidate | Attend New Hire Orientation | Attend Call Center Training | Assign Mentor |
| Background Check | Complete HR Forms | Establish Work Schedule/Shift | Probationary Review |
| Inform Candidate of Rejection | Obtain Badge, Network ID, and Equipment | Activate Call Center Profile | Initiate Performance Plan |
| Activity | Enter Information into Employee Database | | |
| | Request Equipment from IT | | |

*Figure 3-10 Participants collaborate in Blueworks Live to define activities*

During discovery, it helps to recognize that we are discovering a process that *takes place today*. Our aim is to create a model that easily communicates *what* that process is today. During process discovery, it is of little significance to document *how* that process works, on what systems, or the features and limitations of those systems existing today. Sometimes a side conversation on these topics is necessary to get all participants in the room to identify with a specific example, but how the process works is less important.

During process discovery, it is also easiest to start with the Discovery Map view in Blueworks Live (Figure 3-11 on page 65).



ll Center Rep Onboarding As-Is (Playback

- Candidate Selection
  - Hire Candidate
    - Select Candidates
    - Schedule Interview with Hiring Man
    - Take Screening Test
    - Conduct Interview
    - Inform Candidate of Rejection
    - Make Offer
    - Inform Recruiter of Offer Rejection
  - Background Check
  - Inform Candidate of Rejection
  - Activity
- Orientation
  - Attend New Hire Orientation
  - Complete HR Forms
    - Complete I-9
    - Complete Tax Forms
    - Complete Benefits Forms
    - Submit Forms to HR
  - Obtain Badge, Network ID, and Equipm
  - Enter Information into Employee Datab
  - Request Equipment from IT
- Training
  - Attend Call Center Training
  - Establish Work Schedule/Shift
  - Activate Call Center Profile

Call Center Rep Onboarding As-Is (Playback Zero)

| Candidate Selection | Orientation | Training | Probationary Work |
|---|---|---|---|
| Hire Candidate | Attend New Hire Orientation | Attend Call Center Training | Assign Mentor |
| Background Check | Complete HR Forms | Establish Work Schedule/Shift | Probationary Review |
| Inform Candidate of Rejection | Obtain Badge, Network ID, and Equipment | Activate Call Center Profile | Initiate Performance Plan |
| Activity | Enter Information into Employee Database | | |
| | Request Equipment from IT | | |

*Figure 3-11 Start discovery in the Discovery Map view in Blueworks Live*

By capturing the unique activities and placing them under milestones in chronological order, we get a simplified view of the end-to-end process before diving into the details. These details include escalation, routing, and business rules that put decision gateways, splits, joins, and events on the Process Diagram view. Work to get a fairly complete discovery map first before moving on to add detail to the process diagram.

## 3.4.1 Helpful discovery exercises

Sometimes it might be difficult to get participants in discovery started. Maybe there is only one person effectively keying information into Blueworks Live. This process is a collaborative one with all participants at the keyboard. Sometimes there are participants who just cannot get past the screens, or who imagine a new “system” better than the system that they already have. Some of these exercises can help.

### Forgetting about the computers

Imagine that there is a solar flare and that all telephones, fax machines, and computers in the world cease to function. We are left with typewriters, paper, interdepartmental mail (brown envelopes), and a courier service. By modeling a process in these terms, we can more easily dismiss and distinguish between a *system* component from the value-added step the system performs.

### Making no assumptions on implementation

Define the entire business process for a full contextual understanding of the process and all participants and stakeholders. Do not skip modeling a step because another system already does that activity. The activity is still a crucial part of the process.

### Reconciling with real business examples (do not invent them)

Discovery of a business process involves multiple SMEs from different departments, business units, or smaller groups within a business unit. Each person is an expert on the same process, albeit a different type of that process (that is, hiring a college graduate versus hiring an industry veteran). Although all parties might already agree that the process is the same, there needs to be room and patience during discovery to reconcile variations between groups and resolve differences in vocabulary to complete a single process model that accurately represents both groups.

By using a real business example, you avoid making up examples that are unlikely to ever occur. What happens is that participants recall “one time way back when” and produce example cases that are corner cases or exceptions to the rule. Although there is an important time and place for these types of cases, there is no need to design the process application to the exceptions rather than the rule at this level of discovering the process.

### 3.4.2 Recognizing patterns in business processes

From BPMN V2.0m we recognize a business process as a sequence or flow of activities in an organization with the objective of carrying out work. As you learn to identify and discover business processes, you begin to see patterns. Many mistakenly view the application as the process. Software applications and systems participate in business processes, but are not the process. Even a workflow application is only an application and is not a business process.

Some who are new to BPM also mistakenly limit their view of a process to the work performed by a single person. This single-user procedural definition is a type of business process (this is, do this, then this, and finally do this). However, a business process can involve multiple human and system activities, span multiple departments, and have an enterprise impact.

As your BPM analysts learn to recognize common patterns in business processes, discovery and analysis activities with process owners and SMEs become more efficient. Process models improve in clarity by recognizing and reusing common patterns rather than writing new variations of a common pattern. Your process analysis and implementation of the business process also benefits from the recognition and reuse of common process patterns.

There are about two dozen patterns well-documented in prior written works on workflow and BPMN for pattern recognition and reuse in BPMN modeling. These patterns illustrate common interactions among people and systems that occur in the way we work. During process discovery, you recognize the conditions present in the process that can benefit from a common pattern and apply one of these patterns, improving the business process and the model as a communication tool to represent the business process.

Applying a common pattern, parallel split, speeds the process by having the instance travel two parallel paths simultaneously (Figure 3-12 on page 68).

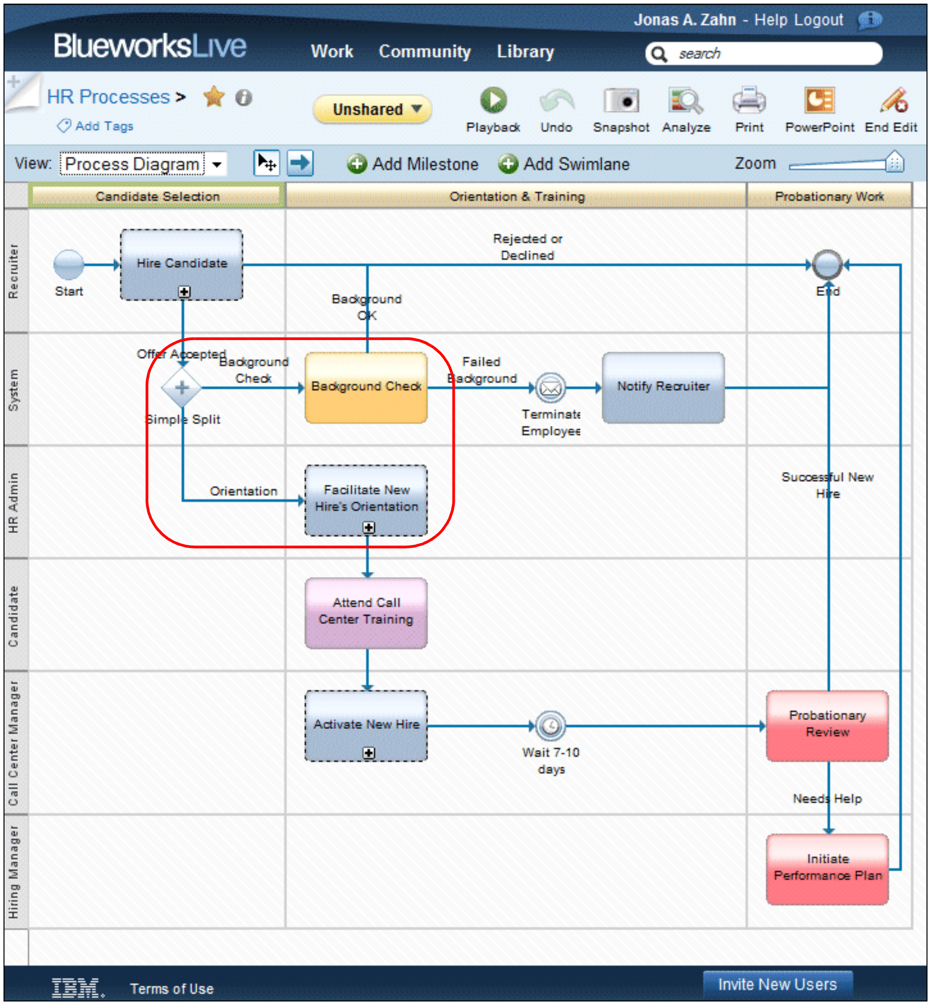


*Figure 3-12 Applying a common pattern, parallel split*

For more information about this topic, go to the following addresses:

- `http://www.bptrends.com/publicationfiles/05-06-WP-BPMProcessPatterns-Atwood1.pdf`
- `http://www.workflowpatterns.com/vendors/documentation/BPMN-pat.pdf`

## 3.4.3 Defining the business case

During process discovery, you continue to improve and refine the business case started during initial assessment. The business case is important to justify the continued investment in analysis and the transition to plan and charter a BPM project, implement a process application, and deploy a first release of the business process application.

### Establishing critical success factors

Your business case describes the critical success factors for the project including sensitivity to time, cost, and scope. In a BPM program with many proposed BPM projects and limited resources, projects with a clear, objective definition of critical success factors might be more likely to get funded, staffed, and deliver successfully on those success factors.

Examples of critical success factors include external time-sensitive factors such as changing regulatory requirements that impact the financial services industry or changes to corporate tax policy that impact capital expenditure decisions and revenue recognition processes. In these examples, the time axis might outweigh success factors that are cost-sensitive or scope-sensitive.

### Identifying key performance indicators

During process discovery, it is important to identify, define, and continuously validate the key performance indicators (KPIs). KPIs are the metrics used to measure process performance. Most managers are already comfortable with cost and time KPIs and typically report on activity-related cost or time metrics in some manner already today.

#### *What a key performance indicator is*

There is some confusion about the relationship between KPIs, metrics, SLAs, reports, and scoreboards. A KPI is a quantifiable measurement that tells you what to measure and the unit of measure. A common KPI is cost. This KPI measures the *cost* of something, and the unit of measure might be US currency (that is, US dollar). For a *time* KPI, the unit might be minutes, hours, or days. A metric is the combination of measures that provides more information but typically requires additional context to be meaningful in the types of decisions that separate a managed process from an unmanaged process.

The operator of an automobile relies on KPIs and metrics to manage the performance of the automobile. Examples of KPIs and metrics include:

- Key performance indicator (unit)
  - Distance (miles)
  - Time (hours and minutes)
  - Fuel (gallons)
  - Temperature (degrees fahrenheit)
  - Pressure (PSI)
- Metric (units)
  - Vehicle speed (mph)
  - Fuel mileage (mpg)
  - Engine speed (rpm)
  - Oil temperature (F)
  - Oil pressure (PSI)

### *Validating KPIs with process owners and stakeholders*

Some process owners have difficulty identifying even one or two KPIs, whereas other process owners can produce a list of 250 KPIs that turn out to be irrelevant. Start first with time-based and cost-based KPIs. Most business processes contain steps (activities) that have service level agreements (SLAs) derived from the time it takes to start or complete an activity or the cost associated with performing the activity.

All KPIs should trace to a decision, a stakeholder who makes that decision, and ultimately to process goals and corporate strategy (Figure 3-13). Arguably, if a KPI does make visible performance data in a way that empowers stakeholders to make decisions, the KPI is irrelevant. If you have a list of KPIs, validate each one by tracing each up the hierarchy shown in Figure 3-13 and identify the stakeholder and the decision made to support corporate goals. Though the KPI might be interesting, if the KPI does not support a decision it is irrelevant.

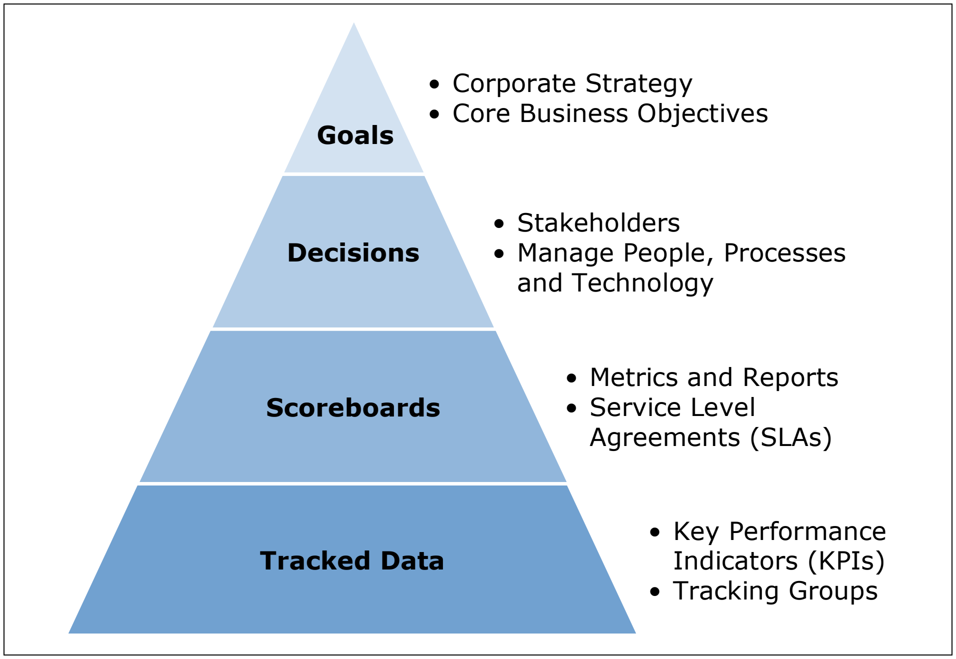

Goals
• Corporate Strategy
• Core Business Objectives
Decisions
• Stakeholders
• Manage People, Processes and Technology
Scoreboards
• Metrics and Reports
• Service Level Agreements (SLAs)
Tracked Data
• Key Performance Indicators (KPIs)
• Tracking Groups

*Figure 3-13 Validate KPIs by tracing upward through the decisions to core business objectives*

Consider the average speed metric recorded by many modern automobiles. You might find it interesting to notice that the average speed (that is, 34 mph) over the life of the automobile might suggest some mix between highway and city driving. Though interestingly, what decision does this metric support? Without a decision that requires this metric, the data collected, and thus the KPI, is irrelevant.

If you have trouble identifying KPIs, go directly to the stakeholders and ask questions to solicit examples of decisions made by the stakeholders. Learn the frequency of those decisions and trace each decision to corporate strategy to validate the decisions. With a list of decisions, the necessary data (KPIs and SLAs) and presentation (scoreboards) begin to materialize.

### Identifying service level agreements

Many people are already familiar with some type of service level agreements (SLAs). As consumers we form expectations for a *level of service* that we receive. For example, when dialing 9-1-1 for an emergency call we expect a live person to answer almost immediately 24 hours per day, 7 days per week, including all holidays and weekends. For other calls, we might expect an answering service after business hours. SLAs are not always derived from the time. An SLA can also reflect the quality of service expected. For example, we would expect an emergency operator to quickly adapt to the caller's language or transfer the caller to someone who can. We expect the operator to not only be awake, but alert, responsive, and proactive in giving direction in an emergency situation.

It is important to identify SLAs in your business processes early and validate them often with stakeholders, SMEs, and process participants. You use these SLAs to *manage* your business process.

#### *What an SLA is*

An SLA is a formally defined expectation for a level of performance. Generally, an SLA is an agreement between two parties wherein a service provider performs work for a customer or stakeholder. In business process modeling, an SLA can be an agreement to complete one step (activity) in the process by the due date. An SLA can span multiple activities and can be based on default or custom KPIs. SLAs empower you to establish a condition for one or more activities that triggers a consequence.

Continuing our previous illustration of KPIs and metrics used to monitor automobile performance, this next figure illustrates how KPIs and metrics are used in SLAs, reports, and scoreboards for managing automobile performance.

With a small set of the previously illustrated simple KPI measurements, the automobile implements a set of complex SLAs, reports, and a dashboard for effective performance management of an automobile. Examples of complex SLAs include:

- Service level agreements
  - Oil life should not exceed 6000 miles. Oil life is measured in total miles since last oil change and a warning light is triggered at 5000 miles.
  - Engine speed should not exceed 7000 rpm. Engine speed is measured instantaneously and triggers a red light at 6000 rpm and kills the engine at 8000 rpm.

  - Oil pressure must not drop below 20 PSI. If pressure is below 20 PSI, a red oil pressure warning light turns on.
  - Oil pressure performance. If oil pressure drops below 20 PSI more than three times in 50 miles, a service engine light turns on and stays on for the next seven engine start events.
- Reports
  - The average speed is tracked in mph.
  - Fuel economy is tracked in actual total miles traveled per gallons used (mpg).
  - Instant fuel economy is tracked by instant speed versus fuel consumption (mpg).
  - Miles traveled including total vehicle miles (odometer) and two trip meters, trip A and trip B.
- Scoreboard (also known as dashboard)
  - Speed (mph)
  - Engine speed (rpm)
  - Engine temp (F)
  - Oil pressure (PSI)
  - Oil temp (F)

Defining an SLA in IBM Business Process Manager involves creating a library artifact (SLA) and giving that artifact a meaningful name and description. An SLA includes a *trigger* and a *consequence*. The trigger by default reads `Whenever the condition is violated` and can be customized to respond to different trends and custom KPIs. The trigger can specify customized values for the trend `whenever` and `condition`. Furthermore, the condition can accommodate custom logic that specifies a KPI and a threshold value to monitor as associated with specific activities. All put together, an SLA might read as shown in Example 3-1.

*Example 3-1 Custom SLA*

```
Name:
   Background Check Failure SLA
Description:
   Monitor the frequency of failed background checks.
Trigger:
   Violated 5% of the time over the last 5 days.
Condition:
   The Background Check SLA for Background Check is equal to false.
Consequence:
   Send email to hiring manager.
```

In our example business scenario, the hiring manager for Call Center Company C wants to monitor the decision to start training new hires without waiting to receive results from a background check. The background check takes several days and the hiring manager hopes to cut those days from the time that it takes to onboard a new hire by running the background check-in parallel. This process could be a costly decision if many new hires fail the background check. For the first release, the hiring manager decides to set up an SLA to monitor the situation and send a notification if 5% of new hires fail the background check over a period of 5 days.

### *Validating SLAs with process owners and stakeholders*

As with KPIs, it is important to validate SLAs with process owners and stakeholders to be sure that they are relevant. An SLA can be configured to send an email notification or start a new process instance (BPD). An SLA is only relevant if the email recipient intends to *take action* or *make a decision*. You might learn that the SLA warrants immediate action at the time that the condition occurs (that is, the minute the task is past due). Although this SLA is valid, the implementation pattern for this type of SLA would involve a timer attached to the activity and a new activity rather than implementing a service level agreement library artifact.

Validate an SLA by maintaining a focus on the business value of the activity, the goal of the business process, and the core business objectives. Trace the SLA and its trigger (based on a KPI), the condition to monitor, the threshold to evaluate, and the consequence to a stakeholder and a decision that helps the process achieve that business value.

## Identifying business pain points

A good business case clearly defines the business pain to build the case for improvement. For many process owners, this step is difficult because defining pain points often involves exposing shortcomings and failures that negatively impacted the business before. For a process owner, this situation could mean admitting to mistakes or taking responsibility for lost opportunity. The success of your BPM program depends on creating a culture of forward-thinking progress and not a culture of blame. A process owner who comes forward with pain points should be rewarded for progress and never shamed for past performance.

All business pain can be summarized as experiencing a *lack of agility*. The key to understanding business pain is to recognize that with visibility and control we can manage our business. Business pain stems from our inability to respond to changes in our business. For example, if there is pain in customer satisfaction levels, the problem is not in our customers, but rather in our inability to:

- See and measure customer satisfaction.
- Make timely adjustments in our business to adapt and correct the behaviors that impact customer satisfaction.

If there is pain in the absence of working capital, the problem is not necessarily that the business is under-funded. Perhaps the business is unable to adequately plan and manage capital expenditures because it does not have the correct information to input into that process. That business pain is a lack of agility in the business.

Business pain can be categorized into three areas:

- Visibility pain
- Management pain
- Continuous improvement pain

Each of these types of business pain can be relieved with tools and techniques in BPM.

### *Business visibility pain*

Pain as experienced by a lack of visibility means that we are unable to respond to events or changing business conditions at all, or not until after the opportunity passes. We cannot take decisive corrective action without valid and timely information. Although we might be subjectively aware of a problem, lack of visibility means that we do not have objective data that we can use to manage our business.

Sometimes visibility pain is the most difficult to identify and define due to the inherent nature of the pain being *invisible*. We might not be aware that there is a problem because we have no exposure to the pain whatsoever.

We cannot manage something that we cannot see. Imagine a fictitious online retailer of candy lollipops. All sales are final and there is no return policy. All sales are online, so the retailer never sees a customer try the lollipops. The retailer has zero visibility into customer satisfaction, and due to good online marketing sales are growing, but the retailer does not realize that there is no repeat business.

Although there are good ways to measure customer satisfaction (repeat business, sales returns, customer surveys, customer support line, online feedback), this retailer has no channel for customer feedback and makes no attempt to measure customer satisfaction. The retailer is experiencing pain, albeit in complete ignorance.

### *Business monitoring pain*

Pain as experienced by a lack of monitoring capability means that while we might be aware of a problem or changing condition, we do not have the tools to take decisive action. Monitoring pain is best illustrated with business activity monitoring (BAM) as a means to watch and manage the daily (or hourly) activity workload of a population of process participants.

Monitoring is often performed by team managers who take corrective action to escalate, reassign, or assist in the completion of tasks. A manager experiencing monitoring pain might know that customer satisfaction suffers or that SLAs are being broken, but does not take immediate corrective action to get back on track.

### *Continuous process improvement pain*

Pain as experienced by lack of agility in continuous process improvement has more to do with long-term ability to adapt and adjust to changes in our business than managing the daily activities in a process. Sometimes there is no shortage of tracked data in a business process, but the stakeholders still do not have the *agility* to make decisive changes in corporate strategy, business resources, or business processes. Part of what makes a business process a *managed* process is having the correct measurements (objective data) to make these types of decisions. The other part is having control to make possible the changes from those decisions.

Continuous improvement pain can be further characterized by having a lack of delivery capability to realize changes in a timely manner. For example, consider a business decision to adjust the behavior of a business process by changing a decision policy for a threshold that requires a management approval step. After the decision is made, it takes weeks (or months) to change and then test and deploy the new solution. In this case, the business pain might best be characterized by the technology, solution design, or delivery methodology.

Business agility in continuous process improvement requires:

- Timely, objective, and relevant information
- Timely decisions for change
- Timely fulfillment of those changes

## Writing the case for return on investment

The business case should demonstrate a positive ROI. Making ROI an explicit part of the documentation helps stakeholders evaluate the business process and decide whether to proceed with Planning and implementation. It is better not to leave the ROI subject to interpretation and lay out the case clearly.

ROI is one way to consider profits in relation to cost of investment. That is, your ROI should make the case for implementing your process instead of doing something else or nothing at all. ROI is a popular metric because of its versatility and simplicity. If a business process project does not have a positive ROI, or if there are other opportunities with a higher ROI, then the business process opportunity should not be undertaken.

ROI is typically expressed as a percentage or ratio of the gains from the investment as compared with the cost of the investment. Figure 3-14 shows that for a business process that might cost $2 million to implement, we expect a gain of $5 million. This situation resolves to a 150% ROI, or for every $2 spent, we realize a return of $3, putting us $1 ahead of where we would be had we not made the $2.

ROI = (Gain from Investment - Cost of Investment) / Cost of Investment

ROI = ($5M-$2M)/$2M = 3/2 or 150%

*Figure 3-14 Formula for ROI*

### *Cost of investment*

As you build your ROI case, you need a rough estimate of the cost to implement the process. With the discovery and analysis completed so far, consider some of the following costs as you build your ROI case:

- Software licenses
- Hardware and infrastructure
- Training
- Delivery (development, test, and deploy)
- Process owner and SMEs participation

In the beginning, you might have trouble figuring out how to address shared costs such as software licenses and hardware/infrastructure investments. As you build your BPM program, the investment per project for these shared resources shrinks considerably. This item is one of the key challenges in selecting a first BPM project. Not only should the project have a high likelihood of success, but the project might require an ROI case that can carry a large portion (or all) of the upfront investment costs that are shared across the BPM program in the future.

### *Gain from investment*

For those process owners who remain solely focused on process *automation*, the gains are often understated in the ROI case. Each of the pain points can be reworded into a value statement with an anticipated gain. It helps to keep a point of view of the cost of *not pursuing* this business process opportunity. Use the pain points previously identified to quantify how agility adds value to the business and support corporate objectives. Common gains that might be hidden in your business process are:

- Value of elimination of rework
- Value of elimination of process variation
- Value of reduced errors (higher quality)
- Value of improved human efficiency
- Value of reduced head count
- Value of new revenue possibilities
- Value of improved customer satisfaction
- Value of employee satisfaction
- Value of employee retention
- Value of activity monitoring
- Value of improved SLA compliance
- Value of improved forecasting
- Value of added visibility for corporate strategy planning

While we do not want to understate business value, we should also be careful not to overstate it either. This situation does not mean that we should “pad” our estimates or be conservative to protect our ROI in the event that our assumptions are wrong. What this situation means is that if we cannot fairly quantify the value of a component of the ROI (that is, customer satisfaction) today, we should still include this component in the ROI, but qualify the value statement as subjective or not measurable today. The new and improved process should give us the tools that we need to quantify these value statements in the future.

## 3.5 Analyzing business processes with Blueworks Live

Before we start planning or implementing a business process, we have an opportunity to analyze what we have discovered. This situation might be the first time stakeholders, process owner, and participants have a comprehensive view of the entire business process. Blueworks Live has tools that you can use to analyze a business process from different perspectives.

### 3.5.1 Analyzing for accuracy

The first objective in process analysis concerns the model itself. Does the model and documentation gathered adequately and accurately represent the business process? Are the activities described to a level of detail that is useful and clearly understood by all? You validate the process model to ensure that the model is both complete and accurate. The model must be a useful representation of the business process. The model should be a good communication tool serving as a common reference easily understood by all parties involved in the discovery, planning, implementation, deployment, and management of the process.

**Accurate BPM:** An accurate business process model is one that:

- Is complete, whereby *every* activity has a complete definition for the activity name, participant, owner, experts, inputs, outputs, customer, supplier, and a user story description
- Is thoroughly and universally understood by the operational business (stakeholders, process owner, subject matter experts, and process participants) and technologists (analysts, solution architects, and developers) as a factual and correct representation of the business process.

### 3.5.2 Analyzing for execution

The process model should also be *executable*. This means that the same process model should be ready for import into the executable runtime environment of IBM Business Process Manager. The model *is* the solution. Process analysis includes reviewing all activities for granularity, context, and completeness appropriate for an executable process.

## 3.5.3 Analyzing for activity granularity

The single most challenging concept in analyzing and documenting a process is *activity granularity* (also known as *task decomposition*). As you analyze your business process for good fit and accuracy, you look for an appropriate level of task decomposition. All tasks at a level, as shown in the discovery map perspective in Blueworks Live illustrated in Figure 3-15 on page 80, should be similar in granularity.

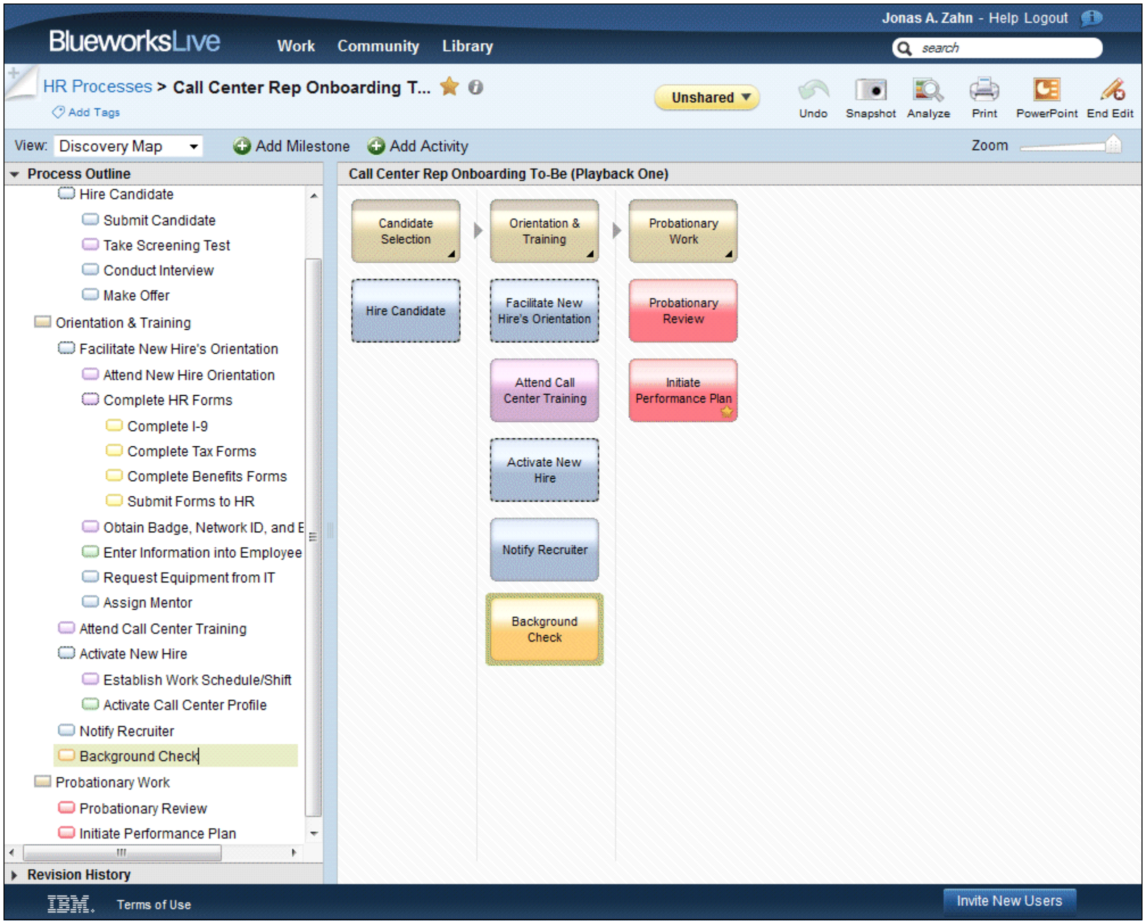


*Figure 3-15 Discovery Map view in Blueworks Live contains activities similar in granularity*

The activities should be similar by some measure, such as size or duration. For example, an activity for a dinner party, such as *wash the dishes*, which might be performed by, at most, two people working together for less than an hour, would not be similar in number of participants or duration if proceeded by the activity *add soap to sink*.

This previous dinner party example illustrates another point about task decomposition. When analyzing the process model for completeness and preparedness for execution, it is important to recognize an appropriate depth in task decomposition. In the dinner party example we could further decompose the activity, *wash the dishes*, into substeps, starting with stopping the sink, turning on hot water, turning on cold water, adding soap, and so on. These steps are far more granular than necessary for a process model that accurately represents the process and is executable. A good practice is to get to a level of decomposition where *one person begins the activity with the intent to finish*.

Typically, an executable process contains a series of activities in which each represents a task that is performed by a single person. The procedural steps *within* each activity do not need to be modeled as *process* steps. This detail can more easily be described with a few words in the user story description.

You also want to analyze for activities that might need further decomposition. A high-level activity that describes work completed by multiple participants each performing a different *step* is probably a subprocess that should be further decomposed to a level of granularity in which each step represents a task performed by a single person.

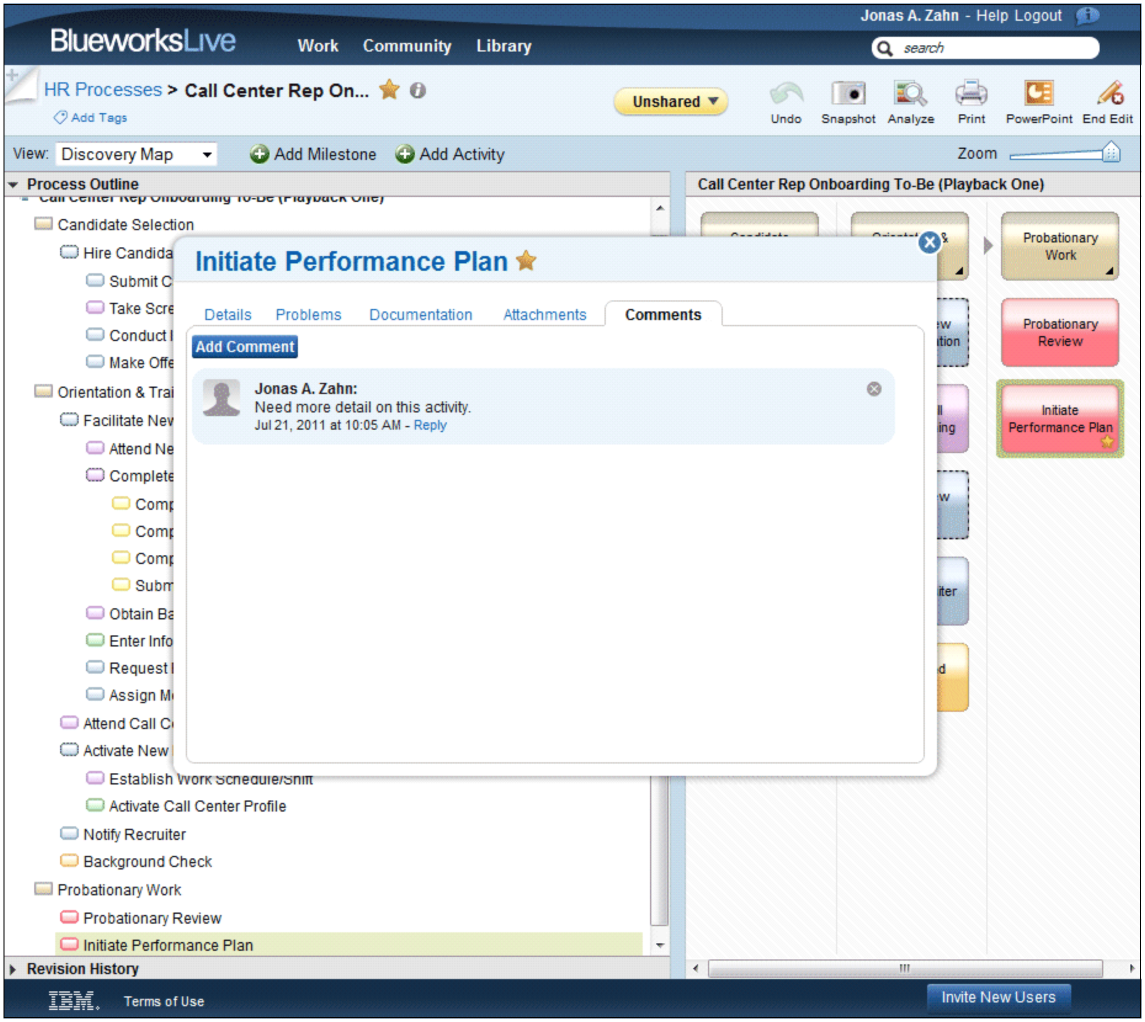
> **Note:** For guidelines about business process modeling, see the topic at the following address:
>
> http://bpmwiki.blueworkslive.com/display/commwiki/Five+Guidelines+to+Better+Process+Modeling

## 3.5.4 Analyzing for opportunity

The second objective in process analysis is to take a deeper look at the business process. Are there opportunities to improve the business process? This situation might be the first time that stakeholders and the process owner recognize an end-to-end business process, view a diagram representation of that process, and think about key performance indicators in the process.

There might already be clear opportunities to combine activities, work on activities in parallel, change owners of activities, or remove non-value added activities to improve the process. There might also be opportunities to define new key performance indicators to start tracking for use in continuous improvement.

For many tasks, you learn from the process owner, experts, and participants that there is tactical pain, but that the As-Is process cannot quantify that pain today. You know that there is a problem, but you do not have the data to prove that there is a problem or justify further effort to fix that problem. In these instances, it is essential to continuous improvement that you identify such activities during process analysis. Highlight the activity (with color), follow it (with a star), and start a discussion thread (comments) (Figure 3-16 on page 82).



Jonas A. Zahn:
Need more detail on this activity.
Jul 21, 2011 at 10:05 AM - Reply

*Figure 3-16 Activity is colored red and starred to emphasize this activity for continuous improvement*

Additional detail should be added in the documentation describing the opportunity for improvement. Analysis should validate the definition of KPIs that will be used to track these problems in the future.

This type of proactive approach to measuring and improving a business process after initial release is essential to transforming your corporate culture. BPM is about continuous improvement. No business process should ever be deployed only once. The value in deploying a business process is not truly realized until release 2 or release 3, when feedback from release 1 is used to improve process performance.

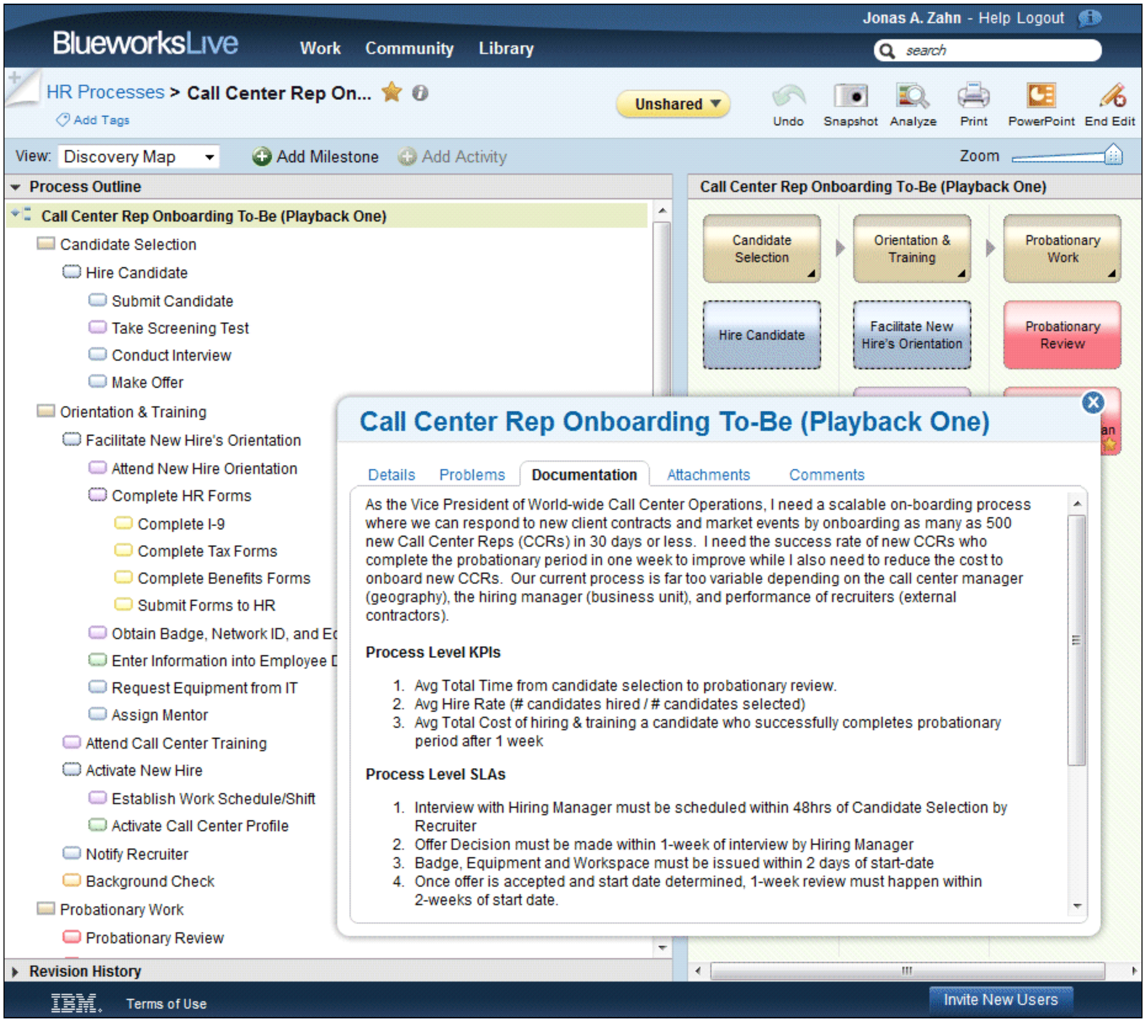
### 3.5.5 Process re-engineering in Blueworks Live

Process engineering (or re-engineering) is a broad definition that includes design, operation, control, and optimization of any type of industrial, chemical, system, biological, or business process. In this book, we address the topic of process engineering and re-engineering in the context of analyzing a business process. Process analysis seeks to answer the following questions:

- Is the process definition complete, or if information is missing is it marked as such for further investigation?
- Is the as-is business process adequately represented with an as-is process model?
- Is there good reason to create a to-be model in addition to the as-is model?
- Are key performance indicators and SLAs for process control defined?
- Is there good documentation highlighting opportunities for continuous improvement?
- Does every activity include adequate detail about who performs the activity and the business outcome?

Process analysis should first validate the boundaries of the end-to-end business process and the business events that can start and end that process. At the highest level in the process model, you should have a clear definition of the process that fits in a few sentences and captures the main goal of the whole business process. The language used in this simplified view begins to expose the KPIs that are used to measure the process performance and roll up to support corporate strategy.

The highest level activity description for the entire process describes level KPIs and SLAs (Figure 3-17 on page 84).



**Call Center Rep Onboarding To-Be (Playback One)**

As the Vice President of World-wide Call Center Operations, I need a scalable on-boarding process where we can respond to new client contracts and market events by onboarding as many as 500 new Call Center Reps (CCRs) in 30 days or less. I need the success rate of new CCRs who complete the probationary period in one week to improve while I also need to reduce the cost to onboard new CCRs. Our current process is far too variable depending on the call center manager (geography), the hiring manager (business unit), and performance of recruiters (external contractors).

**Process Level KPIs**

1. Avg Total Time from candidate selection to probationary review.
2. Avg Hire Rate (# candidates hired / # candidates selected)
3. Avg Total Cost of hiring & training a candidate who successfully completes probationary period after 1 week

**Process Level SLAs**

1. Interview with Hiring Manager must be scheduled within 48hrs of Candidate Selection by Recruiter
2. Offer Decision must be made within 1-week of interview by Hiring Manager
3. Badge, Equipment and Workspace must be issued within 2 days of start-date
4. Once offer is accepted and start date determined, 1-week review must happen within 2-weeks of start date.

*Figure 3-17 The highest level activity description for the entire process describes level KPIs and SLAs*

## As-Is and To-Be (or not To-Be)

Also known as *current state* and *future state* models, you will decide whether your process requires documentation of both as-is and to-be states. You need not always create both as-is and to-be versions of a business process.

## Not To-Be

Typically, we begin process discovery by documenting the As-Is process. This is the process that *happens* today regardless of how the process works today. Discovery does not *change* the process, and implementation does not need to change the process either.

A valid process model *describes* the business process as a series of activities (tasks) carried out by participants (human or system) with little regard for the technology. More simply stated, the process model describes *what* the steps are and not *how* they work. To this end, the original as-is process model could also be the executable business process model with no need to create an additional to-be process model.

If the process model contains information that specifies how activities function (often with technology or system-specific language) that require a new model merely for implementing the process in IBM Business Process Manager, then the model contains the wrong information. This reason is not a good reason to create a to-be version of the process model.

## To-Be versions

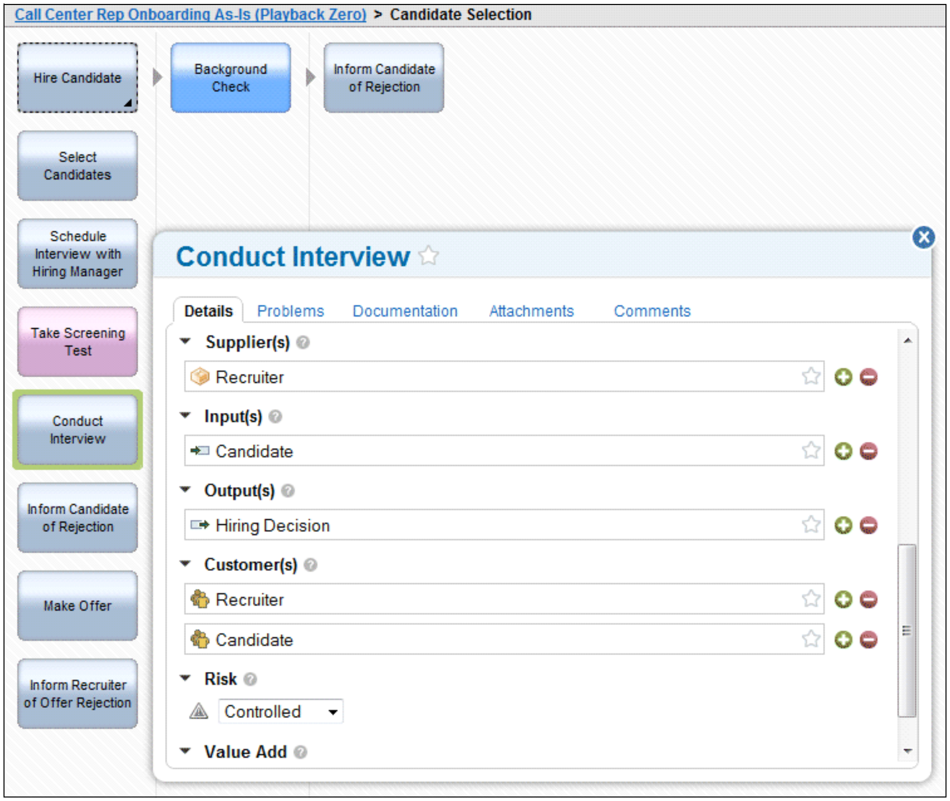
If you are making significant changes to improve the business process, a to-be version of the business process model can be helpful. The pair of models make it easier to collaborate during analysis. Having both the as-is and to-be versions available side-by-side to compare and contrast also improves communication and illustrations for presentations during analysis. Where the as-is version serves to communicate and illustrate what the process is today, the to-be version shows that the process can be improved in its future state.

With Blueworks Live, you can use snapshots to create versions of a process. It might be adequate for your team to take a snapshot of the as-is process before making minor modifications for the to-be state. You can always revert to the as-is process to get a view and compare. If the planned changes are significant, it might be preferable to make a copy of the as-is process and then start modifying the copy for the to-be version while maintaining the as-is process model separately.

## SIPOC analysis in Blueworks Live

SIPOC analysis entails completing a structured definition and analysis of every activity (subprocess or task) to include listing the Suppliers, Inputs, Process, Outputs, and Customers. The purpose of SIPOC analysis is to better understand the transition, or hand off, between process steps or activities. For an activity, the supplier is often the participant of the preceding activity and the supplier's output from the previous activity is also the input for the activity.

In Blueworks Live, there is also the opportunity to expand analysis to include cost, value-added, duration, and problems associated with the activity. The more complete the documentation is, the more useful the model is in analysis mode (Figure 3-18 on page 86).


Call Center Rep Onboarding As-Is (Playback Zero) > Candidate Selection
Hire Candidate
Background Check
Inform Candidate of Rejection
Select Candidates
Schedule Interview with Hiring Manager
Take Screening Test
Conduct Interview
Inform Candidate of Rejection
Make Offer
Inform Recruiter of Offer Rejection
Details
Problems
Documentation
Attachments
Comments

Conduct Interview

Supplier(s): Recruiter

Input(s): Candidate

Output(s): Hiring Decision

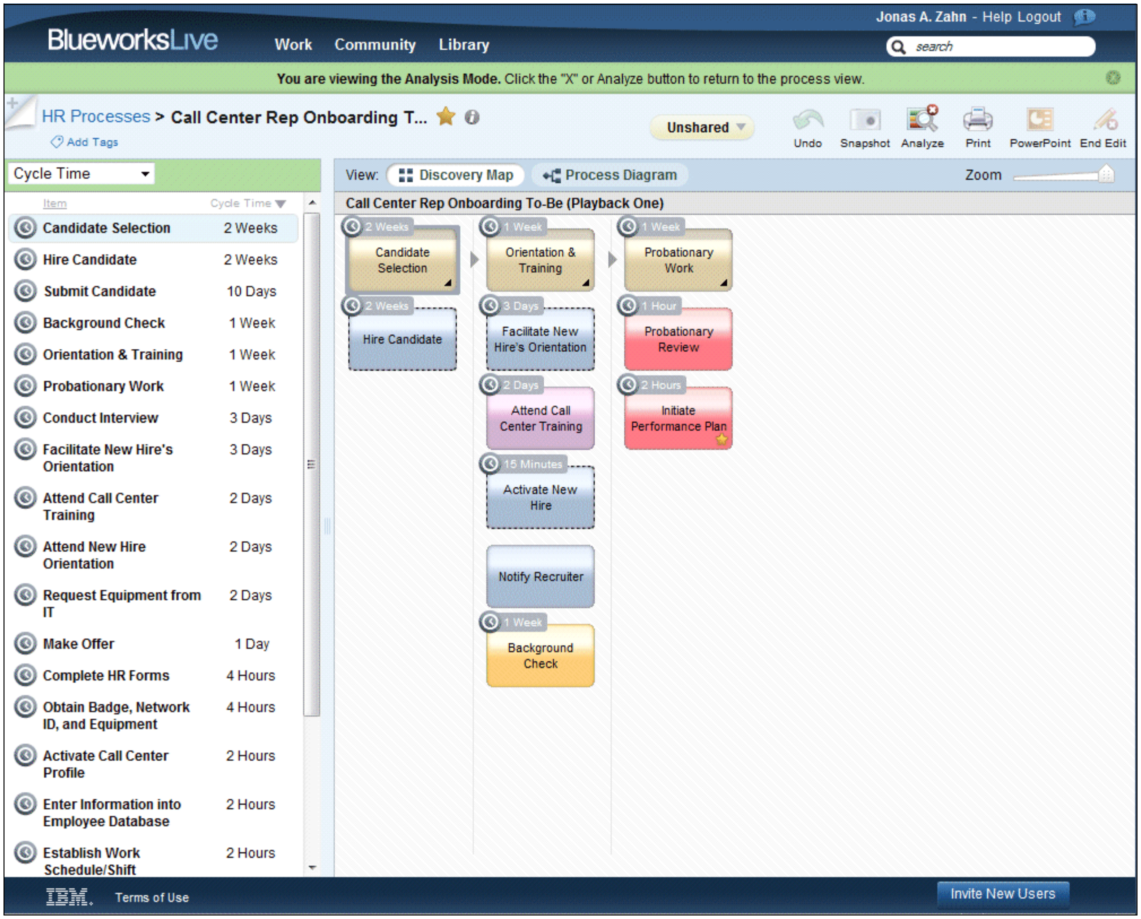
Customer(s): Recruiter; Candidate

Risk: Controlled

Value Add

*Figure 3-18 SIPOC data entry in Blueworks Live helps complete process definition*

Figure 3-19 on page 87 shows the analysis mode in Blueworks Live showing a roll-up view of different metrics.



You are viewing the Analysis Mode. Click the "X" or Analyze button to return to the process view.

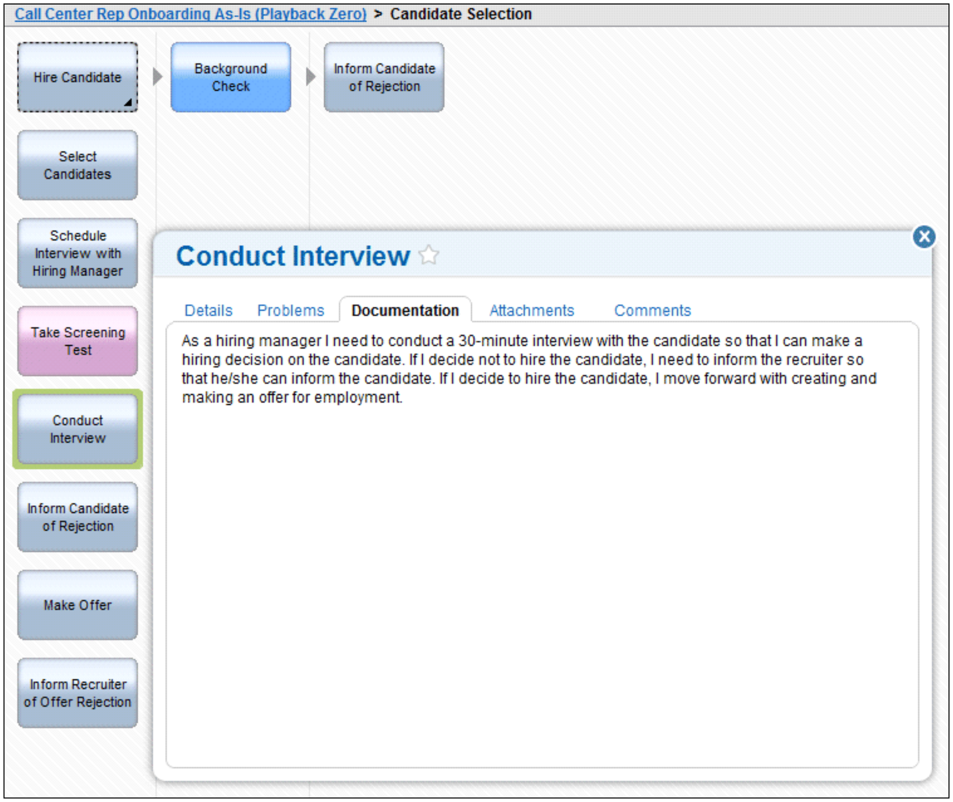
Cycle Time

| Item | Cycle Time |
|---|---|
| Candidate Selection | 2 Weeks |
| Hire Candidate | 2 Weeks |
| Submit Candidate | 10 Days |
| Background Check | 1 Week |
| Orientation & Training | 1 Week |
| Probationary Work | 1 Week |
| Conduct Interview | 3 Days |
| Facilitate New Hire's Orientation | 3 Days |
| Attend Call Center Training | 2 Days |
| Attend New Hire Orientation | 2 Days |
| Request Equipment from IT | 2 Days |
| Make Offer | 1 Day |
| Complete HR Forms | 4 Hours |
| Obtain Badge, Network ID, and Equipment | 4 Hours |
| Activate Call Center Profile | 2 Hours |
| Enter Information into Employee Database | 2 Hours |
| Establish Work Schedule/Shift | 2 Hours |

*Figure 3-19 The analysis mode in Blueworks Live showing a roll-up view of different metrics*

## Capturing user stories in Blueworks Live

Blueworks Live makes it easy to capture user stories in the documentation panel of each step in a process including activities, events, and gateways. Following a standard user story format often makes it easier for participants to start typing and documenting their stories. The user story format also helps clarify who performs the activity, what is performed, and why. The *why* should reiterate the business value and might expose more KPIs (Figure 3-20).


Call Center Rep Onboarding As-Is (Playback Zero) > Candidate Selection
Hire Candidate
Background Check
Inform Candidate of Rejection
Select Candidates
Schedule Interview with Hiring Manager
Take Screening Test
Conduct Interview
Inform Candidate of Rejection
Make Offer
Inform Recruiter of Offer Rejection
Conduct Interview
Details
Problems
Documentation
Attachments
Comments

As a hiring manager I need to conduct a 30-minute interview with the candidate so that I can make a hiring decision on the candidate. If I decide not to hire the candidate, I need to inform the recruiter so that he/she can inform the candidate. If I decide to hire the candidate, I move forward with creating and making an offer for employment.

*Figure 3-20 User story format helps participants define process steps*

A good user story keeps the focus on *what* is required and stays entirely away from *how* that requirement is met, leaving the focus during implementation on delivering business value instead of on specific implementation details. Moreover, process owners and SMEs might not be as informed about different ways to implement a feature as the technology team.

Figure 3-21 illustrates a poorly written user story that specifies implementation details. Writing “I need a web form” does not document what the user needs to do.

> As a hiring manager, I need a web form to enter the results of my interview with the candidate. If I decide not to hire the candidate, I need to email the recruiter so that the recruiter can call the candidate. If I decide to hire the candidate, I need to open an Microsoft Word template and draft an offer for employment.

*Figure 3-21 A bad user story with implementation-specific details*

Further details specifying *email* and *call* limit the solution to specific technology or methods of communicating information from one party to another. The user story could be written much better, as shown in Figure 3-22.

> As a hiring manager, I need to conduct a 30-minute interview with the candidate so that I can make a hiring decision about the candidate. If I decide not to hire the candidate, I need to inform the recruiter so that recruiter can inform the candidate. If I decide to hire the candidate, I move forward with creating and making an offer for employment.

*Figure 3-22 Better written user story*

This user story leaves out all implementation details as to *how* the work is done, but maintains a clear definition of *what* the hiring manager needs to accomplish and why.

The user story format of documenting process steps not only makes it easy to capture important details about the process, but user stories also lend themselves to agile software development methods. User stories keep everyone focused on business value during planning and implementation by refraining from language that is specific to systems or design patterns.

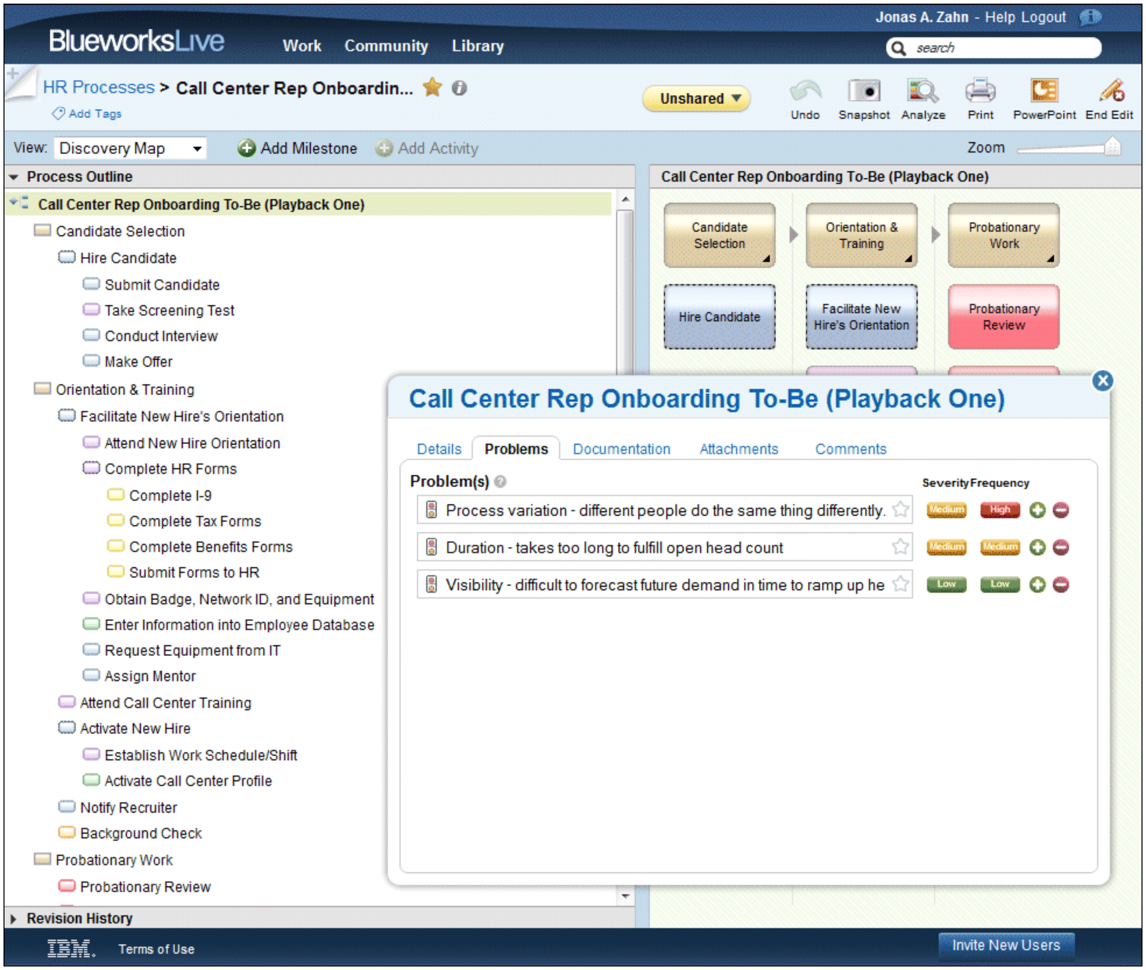
## Tactical pain points

During process analysis, you should highlight and bring attention to areas of the process that exhibit pain in terms of rework, process variation and irregularity, lack of visibility, excessive cost, inefficiency, bottlenecks, errors, or costly corrections. You might begin by tracking problems in activities and later add more detail about the cost, systems, participants, and value-added nature of the activities. This information is crucial to supporting the business case.

Remember, in some cultures, it might be difficult to solicit problem statements from process owners, experts, and participants because these problems might reflect badly on prior performance. For some people, talking about process challenges and known issues in a qualitative or subjective sense is difficult enough. Changing the dialog to look for objective ways to measure those problems or the impact of those problems with KPIs can become uncomfortable or even aggressive for some participants.

**Sensitivity about business issues:** Nobody likes to air their dirty laundry in public. Getting business participants to talk about problems might require sensitivity and a change in corporate culture. Changing the conversation to one about objective measures and continuous improvement might prove to be even more difficult than anecdotal discussions about the problems.

When talking about tactical pain points in your processes, it is important for the process owner to lead the discussion (Figure 3-23). Process owners, with help from BPM analysts, are much more successful in soliciting problem statements from process participants. Process owners can also help keep a healthy dialog focused on continuous improvement and solicit the KPIs necessary for continuous improvement.



Problem(s)

| | Severity | Frequency |
|---|---|---|
| Process variation - different people do the same thing differently. | Medium | High |
| Duration - takes too long to fulfill open head count | Medium | Medium |
| Visibility - difficult to forecast future demand in time to ramp up he | Low | Low |

*Figure 3-23 Document pain points in the process as problems*

### Tactical solution architecture

During process analysis, the BPM solution architect begins forming a solution proposal. This proposal includes a tactical architecture of the business process. At the tactical level of the solution architecture, a business process can be described as *who* does *what*, *when* does it needs to be done, and *why* it needs to be done.

#### *The who (and when): Process participant*

For each task in the business process definition, much care should be taken during process analysis to validate and document who can perform each activity and when that activity must be performed. The *who* is the process participant, but should be elaborated on in the description if there are particular requirements, such as seniority, licensing, or geography. For example, an activity might be performed by a CSR, but only a CSR Level 2 licensed in California can work on certain activities. Elaborate on the *when* if there are particular requirements (or no requirements) for when an activity must be (or can be) started or completed. This information is essential during implementation when designing activity routing policy, due dates, escalation, user calendars, and SLAs.

#### *The what (and why): User story detail*

Your process analysis should validate the user story detail to make sure that the goal and output of the task, a description of what the activity must do, are clearly documented. Do not leave the assessment of business value, the reason why this task must be performed, up for assumption. Make sure that the business value or business impact is also described in the documentation for the activity. Without this documentation, the process definition is incomplete and might miss opportunities for additional key performance indicators and metrics. Without a clear statement of business impact, there might also be inappropriate attention to detail during implementation for this activity.

## 3.5.6 Playing it all back with Playback Zero

Playback Zero is an important milestone in the lifecycle of a business process. For many participating in discovery and implementation, this feature is their first experience with a formal *playback*. This situation might also be the team’s first experience with IBM Business Process Manager and the user experience with the Process Portal. Playback Zero might be the first time that the team experiences the newly designed business process from end-to-end with a real-world business scenario.

**Do not skip Playback Zero**: This first playback is an important milestone and transition point in the lifecycle of your business process.

Playback Zero is also a transition point between discovery and implementation phases in the lifecycle of a business process application. At this juncture, stakeholders, including business owners and technology owners, have a thorough and common understanding of the business process, impact, scope, complexity, risk, and potential ROI if the process is implemented successfully. This situation is an opportunity for stakeholders to decide whether to continue with planning a project and implementing the process or to postpone working on this process in lieu of other process initiatives. Formalizing this transition as a decision opportunity in the lifecycle of a business process is another milestone on the journey to BPM adoption. Visibility and control over which processes to continue working on, and when, is another way in which business stakeholders take control of their business.

Playback Zero is the culmination of the modeling effort to create the to-be process model. The model completed in Blueworks Live can, in a matter of minutes, be imported by IBM Business Process Manager V7.5 and played back in a real runtime process execution environment. The documentation captured in Blueworks Live stays with the executable process model in IBM Business Process Manager. The process can be *played back* with the press of a button, where participants can experience the series of steps (tasks) as a sample business scenario is prompted through the process from end-to-end.

There are a few factors that make playbacks successful. See Chapter 6, “Deploying a process” on page 183, for more details about conducting a playback.

## Process owners can lead the playback

When the process owner leads the playback doing the *showing* and *telling,* the dynamic in the room changes entirely. Rather than the technology or delivery team performing for stakeholders, when the process owner presents the results of the collaborative effort, the entire message has more credibility. There is a good reason for this situation. By having the process owner present his process, the process owner has one more reason to be fully committed and engaged in the discovery, planning, and implementation of *his* process.

> **Quotable:** A poorly articulated message from the process owner carries 10 times more meaning than a well-delivered message from developers.

### Playing back the process from start to finish

Write a business scenario for the playback that can illustrate the process from end to end. Each milestone playback should exercise the entire process. All steps in the process should function, even if stubbed, for every playback. It is reasonable to spend more time on the activities that illustrate the work completed and the theme of the current iteration, but the same end-to-end business scenario should be used to illustrate the full context of the business process in every playback.

### Playing back the process, not the technology

It is important to demonstrate business value in every playback. Stakeholders are interested in the business process and the business value realized by the implementation of that process, and not the technology used to implement the process. Cool features, web services, and clever user interface design are secondary to business value. The best way to keep stakeholders engaged and coming back to subsequent playbacks is to show business value at every playback. TIt is important to not only play back the process from the participants' point-of-view, but also from a manager's perspective, illustrating the scoreboards, reports, and use of KPIs and SLAs to control the process.

### Never skipping a playback

Schedules are not easy to maintain among even a small group of stakeholders. But never postpone or skip a playback, no matter what happens, even if parts of the implementation are behind schedule. These milestones are important. Each playback serves as a validation and feedback opportunity for stakeholders. More importantly, playbacks are an opportunity to market the success and status of a business process implementation. Postponing or skipping a playback jeopardizes the message, and therefore, can risk the success of the project even if the delivery is going well. And lastly, playbacks are good for team morale. Skipping a playback takes away the opportunity for feedback (either positive or constructive), which contributes to positive team morale with reassurance of accomplishment from stakeholders.

## 3.6 Next steps

A successful discovery exercise is useful in producing not only the blueprint for the next major milestone of the project (implementation), but also for aligning the current project to the enterprise value chain by way of a business value analysis. As such, the steps following discovery exist on both the tactical and the strategic level.

The next tactical steps are transforming the user stories into estimation points and a design approach to be used during implementation. The strategic next steps are evaluating how this project impacts the enterprise value chain and preparing executive leadership for that impact, or gaining budgetary approval in context of the impact.

For more information, see *Five Guidelines To Better Process Modeling* at the following address:

http://bpmwiki.blueworkslive.com/display/commwiki/Five+Guidelines+to+Better+Process+Modeling

4

# Planning a BPM project

The promise of BPM is business agility. Business agility is important in a climate of constant change. Changes come from different sources both internal and external to your company, including changes in the market, customers, competitors, climate, politics, and regulations. A successful organization is one that is empowered with the agility to respond to change.

**Estimating and scoping**: If you are reading this chapter because you have an immediate need to learn how to estimate and scope a project, go to 4.3, “Estimating the BPM project scope” on page 130.

Business agility can be achieved by maintaining a constant focus on business value. The value proposition must be self-evident in every decision, from planning a project to managing a BPM program. If there is no measurable business value in an activity, stop doing it. The value you seek should also be in close alignment and traceable to corporate objectives.

As with process discovery, successful project planning depends on a cultural transformation in your organization. This cultural change takes time and you should set an expectation within your organization that the project-to-program journey takes at least 2 years and could take as many as 5 years. This transformation leads you from your current state to a future state where BPM project success is repeatable and not dependent upon individual contributors.

A successful project-to-program transformation means that you demonstrated repeatable success in project planning. It means that you have a BPM program governing project selection, and that you are consistently delivering business value by ensuring your organization is working on the right projects at the right time. Your program provides continuous education and skills development of your people, further empowering the spread of BPM project success throughout your organization. The barrier between IT and operational business has all but disappeared and remains only for purposes of separation of duty. IT participants can easily trace their daily work to corporate business objectives and can describe the business value that results from their effort.

Scaling the success from single project to multiple projects and establishing a BPM program is an enterprise transformation (Figure 4-1).

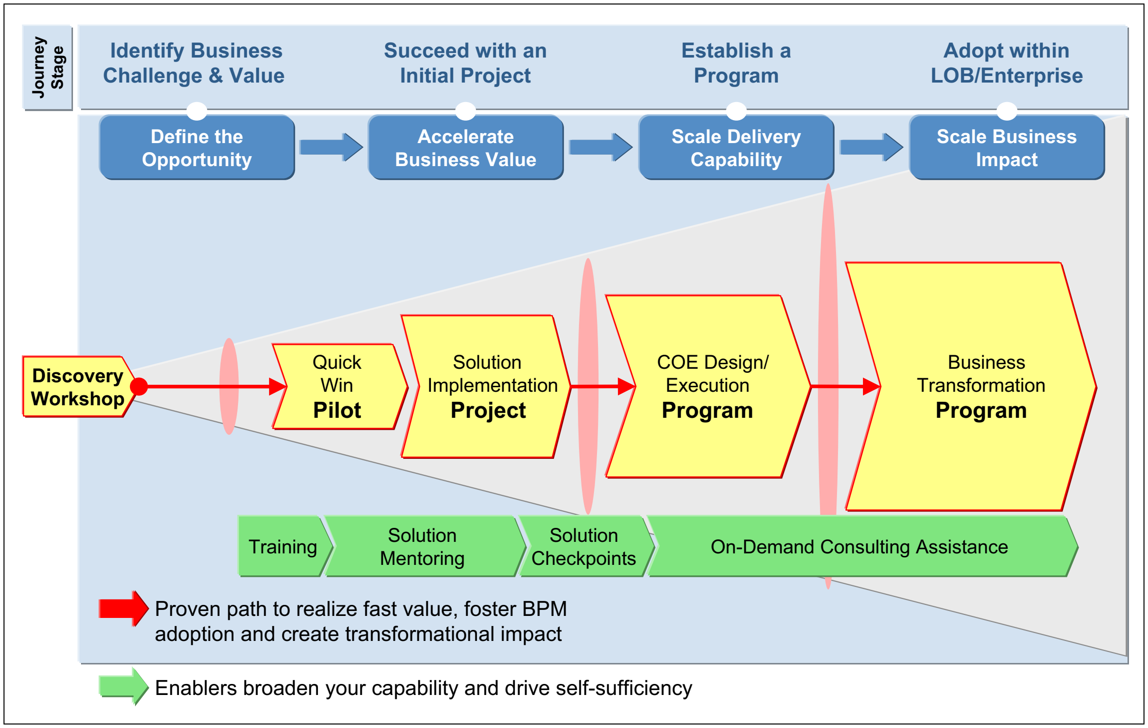

Journey Stage
Identify Business Challenge & Value
Succeed with an Initial Project
Establish a Program
Adopt within LOB/Enterprise
Define the Opportunity
Accelerate Business Value
Scale Delivery Capability
Scale Business Impact
Discovery Workshop
Quick Win Pilot
Solution Implementation Project
COE Design/ Execution Program
Business Transformation Program
Training
Solution Mentoring
Solution Checkpoints
On-Demand Consulting Assistance
Proven path to realize fast value, foster BPM adoption and create transformational impact
Enablers broaden your capability and drive self-sufficiency

*Figure 4-1 Project to program as an enterprise transformation*

BPM Enablement activities such as formal training classes and mentorship lead to self-sufficiency. Do not cut the journey short by trying to start in the middle. You must achieve success with a short “quick win” project before you can expand to multiple projects. You use assets and lessons learned from the quick win project to accelerate BPM adoption. Your first successful projects establish credibility in both technology and methodology and lay the foundation for a BPM program. Only after establishing a program are you ready to create a BPM Center of Excellence that can scale the program to enterprise-wide adoption.

After discovery of a business process reaches a reasonable level of maturity (in Agile Software Development, you never stop iterating with discovery and documentation), the process is ready for you to start planning for implementation. Prioritization, scope, scheduling, and team structure discussions begin and a project is formally started.

This chapter focuses on the Planning phase in the lifecycle of a process in BPM (Figure 4-2). Planning starts toward the end of Discovery after deciding to proceed with initial implementation. Project planning takes place over a period of weeks during Playback 0 and typically culminates with Playback 1. Planning activities continue throughout the iterative process of implementation.

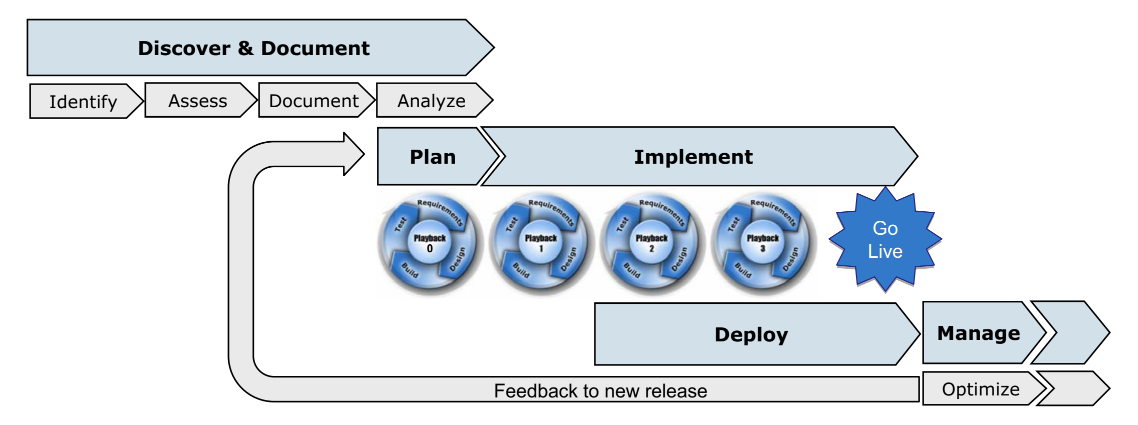


*Figure 4-2 BPM lifecycle*

In this chapter, we describer the following concepts in BPM planning:

- Achieving BPM maturity through skills development
- Agile planning and management for BPM projects
- Estimating the BPM project scope

## 4.1 Achieving BPM maturity through skills development

Whether you are preparing for your first BPM project or finishing your 21st BPM project, skills development for the different roles in BPM is essential to scaling BPM adoption in your organization. A successful organization recognizes the broad range of skills needed in BPM and the value of continuous development as essential elements to BPM adoption and organizational transformation. Scaling from a few successful projects to a BPM program depends almost entirely on the program's ability to attract, educate, and retain people within your organization. Skills development cannot be achieved through educational courses alone. Although initial training classes are a necessary foundation, program-level success can be achieved only through formal mentoring, continuous education, and experience achieved by delivering successful BPM projects.

### 4.1.1 Cultivating skill sets in different roles

Before planning a BPM project or establishing a program, it is important to understand the roles associated with process discovery, development, deployment, and continuous improvement. The quantity and utilization of these roles in our organization differ based on process complexity and size. As processes get larger and more complex, you need more seasoned and experienced staff to successfully implement and manage project delivery. You already have a group of talented individuals that have a mixture of analytical, development, and management experience and who are interested in participating in BPM projects.

This section explains the responsibilities and skills needed for delivering a process and how to empower your team to fulfill the roles required in BPM. As individuals gain experience and achieve success with BPM, you recognize the maturing skill sets and growing responsibilities that IBM identifies with certification levels (Level 1, 2, and 3) described for each role in this chapter (Figure 4-3). Skills development must be planned and is role-specific. Sufficient skills development improves BPM adoption, achieves incremental business value, and ensures BPM Program success.

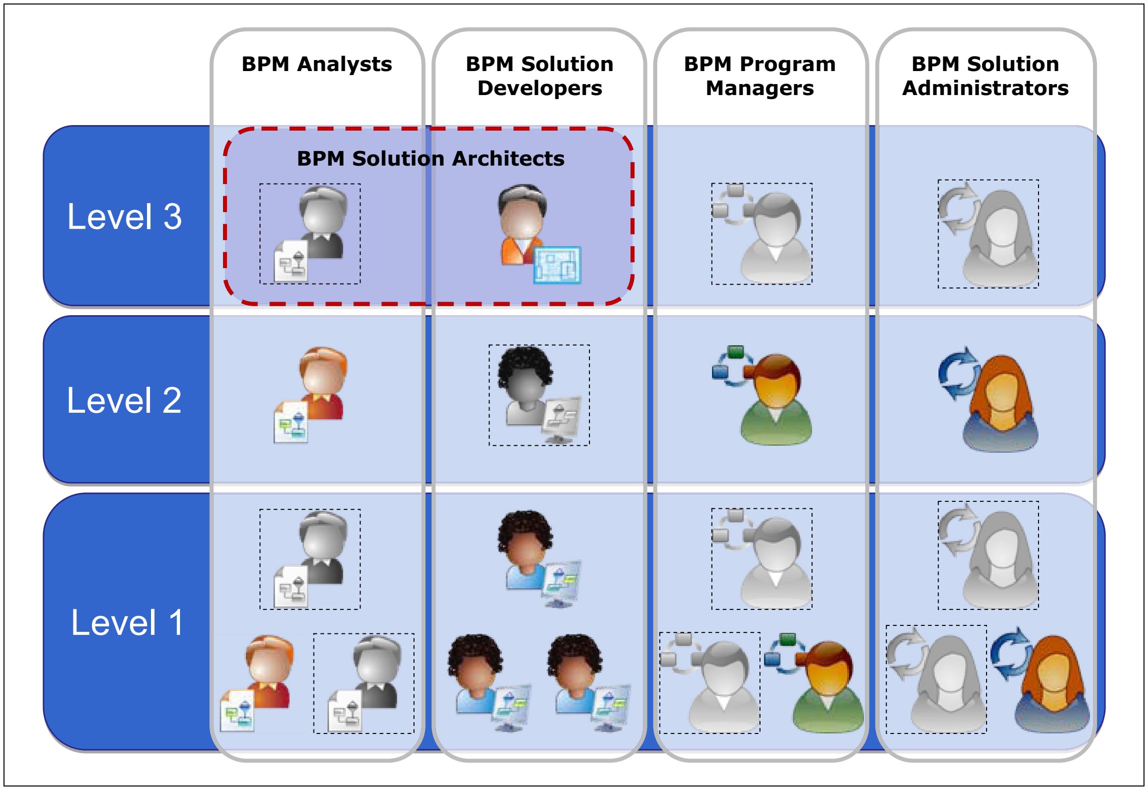


*Figure 4-3 Skills development must be planned and is role-specific*

### Creating a culture of we

Developing a *we* culture is important because BPM projects are highly collaborative with shared work involving several different roles. The roles described are not specific to *IT* or *business*. Although some roles are typically filled by persons with specific IT or business background, there is less *stickiness* in IT or business in the nature of their work, who they talk to, and the tools they use. For example, it is common for a Process Owner or Subject Matter Expert (SME) (lead process participant) to collaborate with a BPM Solution Administrator to plan and carry out the deployment of a new version of a process application. Each role contributes to the conversation and uses the Process Admin console. Although separation of duty is achieved through configuration and governance within the Process Center, such separation is not achieved through boundaries across roles and the tools in the traditional sense in which your staff are accustomed.

Having access to Blueworks Live and Process Designer gives the entire cross-functional team the freedom to review, comment, and make changes (based on appropriate permission and governance policies) to the process. This situation supports the necessary sharing of information, transparency, and the ability to communicate effectively between the roles.

### Encouraging cross pollination of skill sets

It is common for business and IT roles to overlap regularly in collaborative conversations and joint activities. All team members involved in a BPM project should be familiar with, and have access to, the tools where the projects are. This familiarity includes the process model in Blueworks Live, access to the process application in the BPM Process Center, and views into the process authoring environment in IBM Business Process Manager.

Some individuals seek skills development and an interest to perform the functions of BPM roles outside of their current job title or assignment. Such lateral and vertical movement, and overlap of skills, should be encouraged in your organization.

Table 4-1 shows a brief description of the different roles, responsibilities, and skills needed for delivering a BPM project.

*Table 4-1 Roles, responsibilities, and skills for delivering a BPM project*

| Role | Responsibilities | Skills and expertise |
|---|---|---|
| BPM process owner | ► Creates, validates, and prioritizes user stories<br>► Plans, commits, and accepts development iterations (sprints)<br>► Accepts completed user stories or provides feedback for changes<br>► Validates and prioritizes defects/issues<br>► Settles differences among SMEs regarding functionality or priority of user stories | ► Knowledge of agile methodology and ability to facilitate team collaboration in an agile environment<br>► Knowledge of the process being developed<br>► Healthy and persistent mission to keep development aligned with business value traceable to corporate objectives<br>► Relentless push to define/refine the metrics and key performance indicators that encourage transparency, visibility, and encourage process improvement |
| BPM process participant (SME) | ► Provides additional depth and detail on process flows, business policies, and user interface interactions as discovered in user stories<br>► Provides feedback (almost daily) on development work in progress; collaborates with BPM analyst and developers<br>► Defines and refines user stories in a hands-on manner (almost daily)<br>► Conducts the Playback sessions for peer review and stakeholder acceptance | ► Deep knowledge of the process/activities<br>► Expert knowledge of process requirements<br>► Ability to socialize process improvements, compromises, and iterative delivery (agile) concepts to their peers.<br>► Self-motivated and driven<br>► Team leader and a champion among peers (process participants) and a good listener<br>► Collaborative team player and effective communicator with IT (analysts, developers, administrators, and project managers) |

| Role | Responsibilities | Skills and expertise |
|---|---|---|
| BPM analyst | ► Leads process improvement efforts<br>► Expert in process discovery, documentation, analysis, scoping, and optimization<br>► Power user of Blueworks Live and Optimizer<br>► Identifies business case, key opportunities, and ROI<br>► Enforces delivery of KPIs, SLAs, and scoreboards.<br>► Facilitates discussions for compromise (in scope) and agile planning/management discussions | ► Experience with process design, requirements gathering, and facilitation<br>► Recognizes common process patterns; can help keep participants focused on topics that matter<br>► Empowers participants to document their own user stories<br>► Critical analysis and reporting skills<br>► Lean and Six Sigma training or certification |
| BPM program manager | ► Expert in agile (iterative) delivery methodology<br>► Guides estimation and tracking of BPM projects<br>► Manages scope based on value, budget, and resources<br>► Establishes cadence and empowers a cross-functional team to self-organize and deliver working software<br>► Fosters communication and compromise between process owner (business) and delivery (IT) | ► Experience delivering iterative projects and using agile management tools/techniques for projects<br>► Experience managing multi-project (program) roadmaps that are delivered with incremental releases<br>► Facilitates business and IT collaboration<br>► Communicates with sponsor and executive levels of the organization<br>► Actively engages all roles in BPM; encourages leadership<br>► Recognized as a "method expert" by stakeholders and team |

| Role | Responsibilities | Skills and expertise |
|---|---|---|
| BPM developer | ► Estimates work (tasks) to deliver user stories<br>► Implements process flows, services, business logic, and user interfaces (coaches)<br>► Builds KPIs, SLAs, and scoreboards<br>► Implements organization model and task routing policies | ► Processes development and changes leadership experience<br>► Web development skills (JavaScript, JSP, SQL, logic flows, User Interface development, HTML, and so on)<br>► Expert in features of BPMS in the context of process applications and solutions<br>► Expert in best practice recommendations and design patterns in process applications |
| BPM solution administrator | ► Responsible for systems architecture<br>► Designs and implements integrations, custom data storage, and complex data manipulations.<br>► Guides infrastructure design and implementation<br>► Deploys process applications and toolkits to runtime environments (TEST, PROD, and so on) | ► Enterprise software development, specifically OOAD<br>► BPM architecture planning and application service development<br>► Core development skills (J2EE, Java, JSP, SQL, SOAP, XML, XSLT, patterns, advanced logic flows, EAI, and so on)<br>► In-depth experience with capabilities to manage deployments, including role-mapping, environment properties, and deployment options |

| Role | Responsibilities | Skills and expertise |
|---|---|---|
| BPM solution architect | ► Responsible for overall solution architecture of process applications<br>► Leads analyst and process participants (SMEs) in designing the process application<br>► Interprets output of Process Discovery and conceives an "executable" process model<br>► Gains consensus among process participants (SMEs) in vision for the process solution and how best to take advantage of the features of the BPMS | ► Extensive (5+ years) experience implementing process applications<br>► Extensive experience recognizing common process patterns<br>► Ability to visualize opportunities to consolidate and make best use or reusable process components and minimize cost of construction and maintenance of process applications<br>► Good estimating skills in quantifying effort based on user story details |

### 4.1.2 Moving traditional IT roles to BPM roles

Referring to the role-based responsibilities described in Figure 4-3 on page 101, consider typical roles in an IT organization and how each differs from BPM roles. We explore how the gaps can be addressed with education, mentoring, and experience. Some individuals move more easily than others, and not all attempts to transition from similar roles to BPM are successful.

For many people, the hardest part of the transition letting go of old habits and redefining the way we measure individual contribution and success. For each role, the transition requires a new perspective on business value. Each of us should clearly understand our individual, and collective, contribution toward achieving business value. When the outcome of our collective effort results in a process application that delivers measurable business value, we achieve success. Until then, the number of user stories written, or number of pages of documentation signed off, or test cases run, or defects found (and fixed), do not matter.

**Success:** The only measure of success is adoption, and adoption follows business value.

## Becoming a BPM Program Manager

Project and Program Management is a broad field. There are multiple organizations and certifications along with methodologies that each company tweaks or implements differently. Although some core tenets are commonly shared, the execution, inputs, tools, techniques, outputs, and gates vary. When adopting BPM, we find some techniques and methods that work better than others.

### *Considering collocation of the team*

A leading practice for BPM is co-locating the team throughout the duration of a project. Being co-located makes it easy for the project to adjust based on meaningful feedback and changing priorities while also engaging in conversation to ensure what is being developed meets the goals. This one activity of increased collaboration is radically different from how most projects are run today. Although having a shared space or other method of keeping the team together is key, it is also important to ensure that the shared space does not become a meeting room. The development team needs a place where they can concentrate to create the designed functionality.

### *Embracing agile methods*

The promise of BPM is business agility. Scaling BPM adoption means extending the capability to process owners to make regular and decisive changes, thus adapting your business to BPM with the support of objective measurements and current performance data. In order for our agile business to respond to change decisively and timely, we must be capable of rapidly measuring, deciding, and changing the processes that support our business. It is therefore necessary to adopt an agile software development and delivery methodology that can govern regular and frequent change to our process solutions.

Most software engineers, project managers, and development leads have been exposed to agile methodologies. Many of these people have some experience in a company that is beginning to adopt or has adopted agile techniques. Having skills in agile approaches, like SCRUM, XP, and OpenUP, is helpful when moving to manage BPM projects.

### *Focusing on business value*

Beyond looking at phases, dates, scope, assignments, risks, and such elements, managing BPM projects is about having a constant focus on business value. The most successful BPM program managers have the capability, experience, and comfort to work in a BPM analyst capacity. These BPM program managers do not replace BPM analysts, but complement the analyst role and bring forward-thinking when assembling work into themes that can be evaluated and compared when creating a process roadmap or a process inventory. These elements are assembled and maintained by the BPM program manager with input (as part of governance) from stakeholders, process owners, and others for establishing a backlog that can be executed based on company strategy or tactical market decisions. As you can imagine, critical thinking with a focus on business value is key when guiding governance decisions.

### *Succeeding with a hands-on approach*

To guide governance decisions and help with project planning and tracking, BPM program managers must have knowledge about how processes are architected. They do not need to be expert BPM developers, but should be familiar enough with BPM to review and play back the process independently. This familiarity helps program managers answer questions associated with effort remaining and solution completeness. This familiarity is especially important when estimating project elements or sizing for a program. Being familiar with relative sizing estimation, like story points, is a key technique that helps program managers easily mature project schedules from estimations to predictive tools. When delivering multiple projects, skills in forecasting and relative sizing of projects to each other becomes more important. A program manager's hands-on approach improves accuracy and confidence in planning, achieving business value, and determining the priority needed for BPM governance.

Developing the skills to manage a BPM program takes time, but it is important to think big, start small, and scale fast. Some steps you can take to become a strong BPM program manager are as follows:

- Be familiar with agile software development techniques. The simplest action is to take a course, read a few books on , and focus specifically on user story estimation, release planning, and iteration planning.
- Understand the difference between the roles played by a program manager in agile methodologies and a program manager in more traditional waterfall methodologies. In agile methodologies, the program manager actively participates in planning and management activities while creating a collaborative environment for self-managing cross-functional teams to stay focused on building working software.

- Get hands on with IBM Business Process Manager and Blueworks Live by taking courses associated with process analysis and modeling, process implementation, and BPM project management. Create an environment that encourages continuous education for all roles.
- Start a project with an experienced mentor.
- Get certified in agile software development and in IBM program management.
- Join the IBM Business Process Manager Community Wiki at the following address:

  `http://bpmwiki.blueworkslive.com`
- Deliver a few processes while continuing to improve already delivered ones.
- Continue expanding your agile knowledge and further your certifications; be a mentor or teacher.
- Continue expanding your education with business process optimization and BPM program management courses.
- Participate in establishing a BPM program.

### *Thinking big, starting small, and scaling fast*

A single program manager can typically manage two small to medium sized projects effectively. As the complexity and size of the team increases, a manager at 50% time is not enough. A first-time project in BPM, even a small one, should also have a dedicated program manager at 100%. It is important when beginning the BPM journey that the BPM program manager role is not regarded a Project Administrator that can span many projects. If the manager is staffed in such a fashion, then it is unlikely the manager is able to garner enough details to effectively manage the project, understand the value being delivered, or guide the necessary decisions to facilitate change.

One of two things happens: Either the project goes “unmanaged” and suffers or ultimately fails to be agile or other individuals on the project take on the role and responsibilities of the program manager and are distracted from their tasks at hand. Although you want to immediately establish a BPM program, growing from architect/manager to BPM project manager and then to a BPM program manager is directly tied to trying a BPM project, deploying multiple processes, building a program, and transforming the business.

## Becoming a BPM Analyst

Many organizations have a small army of business analysts, and most have at least a few individuals who carry this title. Business analysts are familiar with the business and know how to identify business needs and translate those needs into objective and specific requirement specifications. For some business analysts, their ultimate achievement is a thorough and detailed analysis resulting in an extensive software requirement specification (SRS) document.

Some business analysts take this dedication to document a step further and elaborate on the newly defined requirements with use cases, use case relationship diagrams, entity state diagrams, and supporting data analysis. The pivotal transition for this business analyst is changing the focus on delivery of documentation to the delivery of *working software* that achieves business value.

Developing the skills to analyze a BPM process takes time. Some suggested steps you can take to become a strong BPM analyst are:

- Be familiar with Six Sigma, Lean, and agile software development techniques. The simplest action is to take a few courses, read some books and focus on process improvement and re-engineering techniques.
- Get hands on with IBM Blueworks Live and Business Process Manager by taking courses associated with process analysis and modeling (WB731) and process optimization (WB007).
- Start a project with an experienced BPM analyst.
- Get certified in Six Sigma and as an IBM Business Process Manager Analyst (Test 000-170).
- Join the IBM Business Process Manager Community Wiki.
- Expand your knowledge through project-based experience and education.

### *Taking requirements to the engineers*

Sometimes the business analyst is a go-between or interpreter that bridges the IT/Business divide. In these environments, the business often does not collaborate directly with engineers and developers. Some business analysts emerge from a SME background with experience in the operational business. Others are systems analysts that emerge from an IT background. Some business analysts achieve Six Sigma Black Belt Certification and are familiar with multiple techniques to analyze processes, document methods and procedures, and dive into root cause analysis. If your organization develops or maintains systems, you have analysts congregating in conference rooms or working at their desks generating volumes of documentation that people try to review and sign off.

**Working software:** The pivotal transition for this business analyst is changing the focus on delivery of documentation to the delivery of *working software* that achieves business value.

### *Focusing on business value*

How do these types of business professionals successfully transform themselves into BPM analysts? You might be reading this book because you are one of these business analysts. Let us guide you in moving to the role of BPM analyst.

BPM analysts are experts at creating a collaborative environment where process owners and SMEs discover, document, and analyze their business processes. A BPM analyst focuses more on facilitating and less on documentation. These collaborative sessions are described in Chapter 3, “Process discovery” on page 41. The BPM analyst facilitates these sessions with the process owner, SMEs, and key process participants. Learning to conduct these collaborative sessions starts with taking courses in process analysis modeling with Blueworks Live. Continued learning in process modeling and process discovery comes with experience and mentoring. A BPM analyst learns best practices, such as how to document process elements, create self-contained and reusable subprocesses, streamline common processes, reduce exception paths, optimize processes, write user stories, and perform supplier, inputs, problems, outputs, and customer (SIPOC) analysis.

### *Using experience in Lean and Six Sigma*

BPM analysts often have a background in Six Sigma and use their process and data analysis experience to identify process problems before moving into future state design innovation. They know how to prioritize and associate improvement opportunities and help guide the team in determining the future state of the process. Learning to do this type of activity dovetails with clearly defining the success criteria for a business process implementation. Learning to do such work, possibly by taking Six Sigma courses, is critical to detecting inefficiencies and improving process quality.

### *Adapting to agile methods*

BPM analysts are advocates for agile methodology. Those analysts with a solid track record defining business processes for agile delivery are the most successful. They understand how to define the future state process at various levels, knowing where to deep dive in analysis and share the vision of phased releases based on business priorities and critical success metrics. The skill and experience to identify business value and deliver that value in a series of short iterative releases helps establish a process roadmap and build an inventory of business processes.

Learning how and when to progressively elaborate large process blocks into smaller blocks (like user stories) along with clustering them into meaningful units based on value is key to successfully planning process delivery.

#### *Creating transparency with metrics, simulation, and process optimization*

The BPM analyst translates business needs and requests into process documentation that expose key performance indicators (KPIs), service level agreements (SLAs), and the resultant metrics. The BPM analyst ensures that the appropriate metrics are included in each release and that the business process is meeting business needs, like regulatory or improvement goals. Defining key process metrics and their associated reporting goals is essential in the business justification for continued process implementation projects. Building a return on investment (ROI) case is an important skill necessary when presenting a proof of concept and business case to stakeholders for future projects. Creating credible process optimization scenarios with both historic data and simulated data means that BPM analysts must learn how to use the Process Designer authoring environment and the Process Optimizer tools.

#### *Implementing processes with aid of a BPM analyst*

For the traditional business analysts, their work typically finishes when implementation begins. For the BPM analyst, there is no transition of requirements or specifications to the developer. The BPM analyst is needed for the duration of the implementation of a business process. For most small to medium sized processes, a BPM analyst is most often used to start the project with process discovery sessions and lead the team through the first two iterations of implementation (Playback 0 and Playback 1). If the Process Owner or SMEs are able to dedicate enough time to manage and elaborate on changes to the process, then the BPM analyst might become involved part-time. For medium to large sized processes, a BPM analyst is needed throughout the project to prioritize user stories, continue analyzing process details, help the team collaborate on user story details, and participate in planning, committing, and accepting development iterations. As your organization matures to a BPM program conducting several large projects concurrently, you have multiple BPM analysts distributing their time across several projects at various stages in the project lifecycle.

### The emerging BPM solution architect

If your organization is like most, you already have many individuals with *architect* in their job title: software architect, enterprise architect, database architect, solution architect, and so on. A BPM solution architect might move from one of these other roles. It is important, however, to recognize that a BPM solution architect is not just another senior architect or a senior Java, .NET, or other application developer.

Like the other BPM-specific roles, the BPM solution architect also maintains a particular allegiance to business value. Although familiar with technical subjects, from SQL to Ajax, from SOA to JMS, and from design patterns to agile methodology, the BPM solution architect is more often found collaborating with the process owner and SMEs. He collaborates with these people to better understand the business process, its key performance indicators (KPIs), its service level agreements (SLAs), and how that process might be best implemented.

### *Moving to the role of BPM solution architect*

The BPM solution architect typically moves from a senior role in engineering or development with five or more years of experience in programming and application design. Some BPM solution architects move from a business analyst role with deep technical experience. The BPM solution architect might have a background in any number of programming languages, including Java, .NET, C++, PERL, COBOL, or mainframe computing. These individuals might spend two to five years in the role of BPM developer or BPM analyst as they mature in achieving certification levels 1, 2, and then level 3.

A level 3 BPM developer or analyst can fill the role of BPM solution architect. The keys to a successful transition to the role of BPM solution architect include the following items:

1. Learn to assess, prioritize, and measure business value in all aspects of solution delivery.
2. Become both practitioner and evangelist for agile methodology.
3. Establish trust among business stakeholders, process owners, and process participants while maintaining influence within IT.
4. Achieve both breadth and depth in knowledge of every component and the many capabilities of the BPMS (IBM Business Process Manager V7.5).
5. Practice and teach BPM best practices, common solution design patterns, and business process modeling guidelines.

### *Becoming a BPM solution architect takes years of experience*

The BPM solution architects emerge from your program over time. Even those architects with five or more years of experience in software programming or architecture spend another five years in BPM before achieving a level of success and experience to take on the responsibilities of a BPM solution architect. For many, the most exciting part about becoming a BPM solution architect occurs when the team achieves significant business value through short iterative delivery cycles. These achievements are recognized by business stakeholders and celebrated among members of the IT team. The savvy BPM solution architects are the ones who quickly align themselves with business stakeholders to better understand business value and corporate objectives. They use their experience in IT to better serve the business.

## The sysadmin in transition to BPM solution administrator

Of all the roles named in a BPM project, the role of BPM solution administrator is most like a traditional IT systems administrator. Nonetheless, the BPM solution administrator, like the other roles, shares a healthy disregard for technology when it comes to being committed to achieving business value in short iterative delivery cycles. The BPM solution administrator accepts the responsibility of frequent deployments of new versions of process applications, monitors the web of dependencies among toolkits and process applications, manages the risk associated with regular and frequent deployments, and provides both transparency and governance. The administrator does these tasks so that process owners and stakeholders can collaborate in managing that risk.

### *Building experts in the BPMS infrastructure*

The BPM solution administrator has deep knowledge of the Process Center and runtime environments. The administrator knows how each environment is configured, installed, deployed, secured, supported, backed up, and recovered. A senior (Level 3) BPM solution administrator might lead a team of systems experts to design and install IBM Business Process Manager V7.5 and all of its components in your organization's IT environments.

### *Specializing in system integrations*

A good BPM solution administrator is also familiar with the Process Designer and the same tools used by the BPM developer. The BPM solution administrator often takes on development tasks to build the more technically challenging integration components for connecting the BPM process to the many different systems in your IT environment. BPM solution administrators often have years of prior experience in Java or .NET development and are familiar with SOA and related integration technologies, including web services, Java messaging, and database connectivity.

### *Joining the team as a BPM solution administrator*

Succeeding in the transition to a BPM solution administrator is no small undertaking. This task is not something that can be achieved after taking five days of training on the software. The responsibilities of this role almost exclusively attract individuals from other system administrative roles in IT. This role is not for an administrator who prefers the command prompt over a graphical user interface. This role is also not for a Linux administrator who prefers racks of servers over a boardroom of process participants drawing on the whiteboard and sequencing activities in Blueworks Live. The BPM solution administrator is as much a team player in the delivery team as the BPM analyst and BPM solution architect.

## 4.1.3 Building a BPM program for staffing BPM process projects

Having learned about the four key roles in BPM, you might already have identified candidates in your organization to transition to these roles. Next, let us illustrate how these roles come together to create a cross-functional, self-directed, and high-performing team to deliver a business process. Let us also show you how the team makeup might change based on the size and complexity of your process projects. As we illustrate these team compositions for different process projects, keep in mind that your teams are composed of individuals with different levels and types of experience.

### Maintaining a focus on continuous learning

As you form your teams, try matching junior and senior people to create a rich learning environment. Consider matching different levels of experience across functions. For example, a Level 3 BPM program manager can mentor Level 1 BPM developers on one project while a Level 3 BPM solution architect can mentor a Level 1 BPM program manager on a different project. The key is keeping a focus on continuous learning as you compose your teams; each individual project is an investment in your overall BPM program. Although assigning senior level individuals to a single high-priority project might seem a good idea, this action could be a detriment to the overall BPM program and ultimately have a negative impact on BPM adoption.

### Measuring experience with certification levels

As with growing new skills of any type, mentors are important. A team of brand new talent without a mentor not only takes longer to learn and longer to deliver results, but these individuals certainly suffer after developing bad habits. Pairing new team members with experienced contributors is important. In making these pairings across your BPM program, you need a way to measure experience and individual success from achievements.

IBM uses the certification levels shown in Table 4-2 to measure individual experience and their achievements. Each new level can be attained only through a combination of formal training, project-based experience, peer assessment, and a certification exam. These same criteria are used for all IBM Business Process Manager delivery resources and IBM Business Partner resources when measuring experience, skill level, and achievements in BPM.

*Table 4-2 Certification levels*

| **Level** | **Experience and achievements** |
|---|---|
| Trained | ▸ Completed five days of classroom (or online) training materials.<br>▸ Trained on core IBM Business Process Manager V7.5 product and methods.<br>▸ Familiar with agile terminology.<br>▸ No certifiable experience deploying a production process application. |
| Level 1 | ▸ Completed at least one year of project-based experience (or equivalent three months of intense boot camp training for college graduates).<br>▸ Performed job responsibilities under supervision of a Level 2 or Level 3 mentor.<br>▸ Certifiable experience implementing and deploying at least one process application to a production environment.<br>▸ Completed Level 1 certification exam. |
| Level 2 | ▸ Completed around three years of project-based experience.<br>▸ Performed job responsibilities under supervision of a Level 3 mentor.<br>▸ Certifiable experience implementing and deploying three or more process applications to a production environment.<br>▸ Experienced in mentoring Level 1 resources.<br>▸ Completed Level 2 certification exam. |

| Level | Experience and achievements |
|---|---|
| Level 3 | ▸ Completed at least five years of project-based experience.<br>▸ Performed job responsibilities at the program level (multiple teams/tracks or multiple concurrent projects).<br>▸ Certifiable experience implementing and deploying five or more process applications to a production environment.<br>▸ Experienced in mentoring Level 1 and Level 2 resources.<br>▸ Recognized as a BPM evangelist and BPM thought leader by peers. |

## Measuring achievement with project-based experience

IBM offers certification exams for several of these BPM roles by level. Passing a certification exam alone does not qualify an individual's readiness to perform at that level; project-based experience and a proven track record of achievement does. Although professional experience attained before joining a BPM program can fast-track some individuals in these certification levels, there is a minimum measure of BPM experience that should be achieved before recognizing these levels. The time it takes your team members to reach these achievements varies as follows based on an individual's prior experience and commitment to learning BPM with project-based experience:

- Level 1 (3 months to 3 years)
- Level 2 (2 years to 6 years)
- Level 3 (4 or more years)

As with skills development of any kind, there are individuals in your organization that do not achieve levels beyond Levels 1 and 2. This fact should not be alarming. For those few individuals that do achieve Level 3, they are the thought leaders and the mentors to help guide the project teams composed of Level 1 and Level 2 team players.

## Forming teams for BPM process projects

Here are examples of delivery team members, including the BPM analyst, BPM developer, BPM program manager, and BPM solution administrator. These members illustrate the experience, quantity, and time commitment of the different roles for process projects of varying levels of complexity from small to large. In 4.3, “Estimating the BPM project scope” on page 130, we explain how to characterize a process project in terms of low, medium, and high complexity.

### *Keeping teams small and cross-functional*

There is a tendency in many organizations to charter, plan, and manage larger projects. Keep in mind these words: think big, start small, scale fast. There is significant research supporting agile methods that keep project teams small (three to six individual contributors paired with stakeholders, a process owner, and SMEs as needed). Teams larger than six individuals should be broken into smaller project teams, or *tracks*, each concentrating on a part of the solution that can be logically separated from the other. There are dependencies, and a BPM program manager and BPM solution architect help manage dependencies on a larger and more complex project with multiple tracks. They key is to remember that smaller teams with six or fewer individual contributors outperform larger teams.

### *Keeping process projects small*

The same thinking that applies to small cross-functional teams also applies to scoping process projects. Keeping the scope small enough for 90 - 120 day release cycles for a single team of five individual contributors yields more business value than a single team of ten or more individual contributors working on a multimonth (or multiyear) release cycle. Break up larger projects into smaller, iterative projects. Remember, think big, start small, scale fast.

### *Staffing a low complexity process project team*

Consider the team illustrated in Figure 4-4 for a low complexity project.

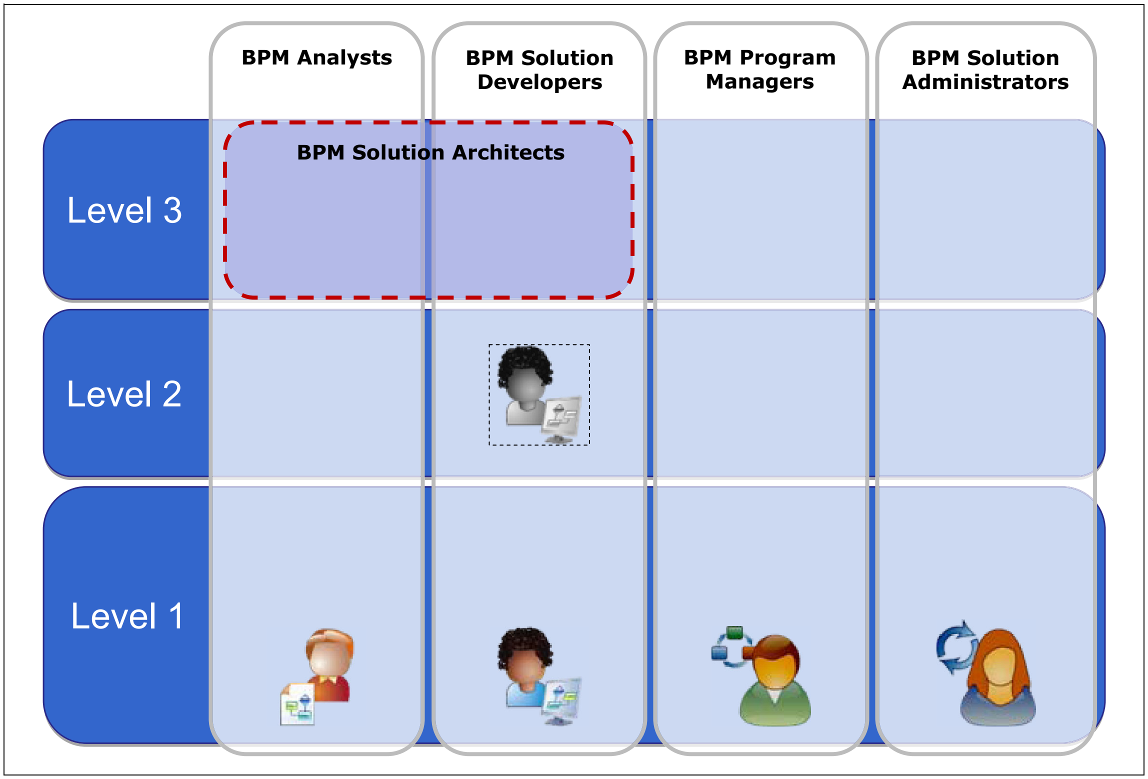


*Figure 4-4 A low complexity process project with a small team of five people lead by a single Level 2 BPM developer*

In general, a low complexity process team includes one of each role and is purely focused on delivering a process project in less than three months. A single Level 2 BPM developer can lead and mentor a small team with a total of five individual contributors. The nature of the project might involve few, if any, system integrations, low complexity in scope with a single business process definition, and no subprocesses with 15 or fewer process steps. The user interface requirements might be low in complexity, leaving much of the implementation details as fairly straight forward. A low complexity project such as this one is an excellent candidate for a single leader/mentor and a small team of Level 1 or recently trained individuals.

For a low complexity process project, the BPM analyst might be full-time during the Discovery phase for 2 - 4 weeks leading up to Playback 0 or Playback 1, and then part-time on a limited as-needed basis for further analysis. The same is true for the BPM solution administrator. The administrator might need to be involved only on a part-time basis for 2 - 3 weeks during the latter half of the project to assist with one or two system integrations and the deployment planning for the process application. The BPM program manager and BPM developers should be full-time. For a low complexity project, a senior BPM program manager and BPM solution architect could also participate in a limited part-time capacity to mentor the team.

### Staffing a medium complexity process project team

Consider the medium complexity process project shown in Figure 4-5.

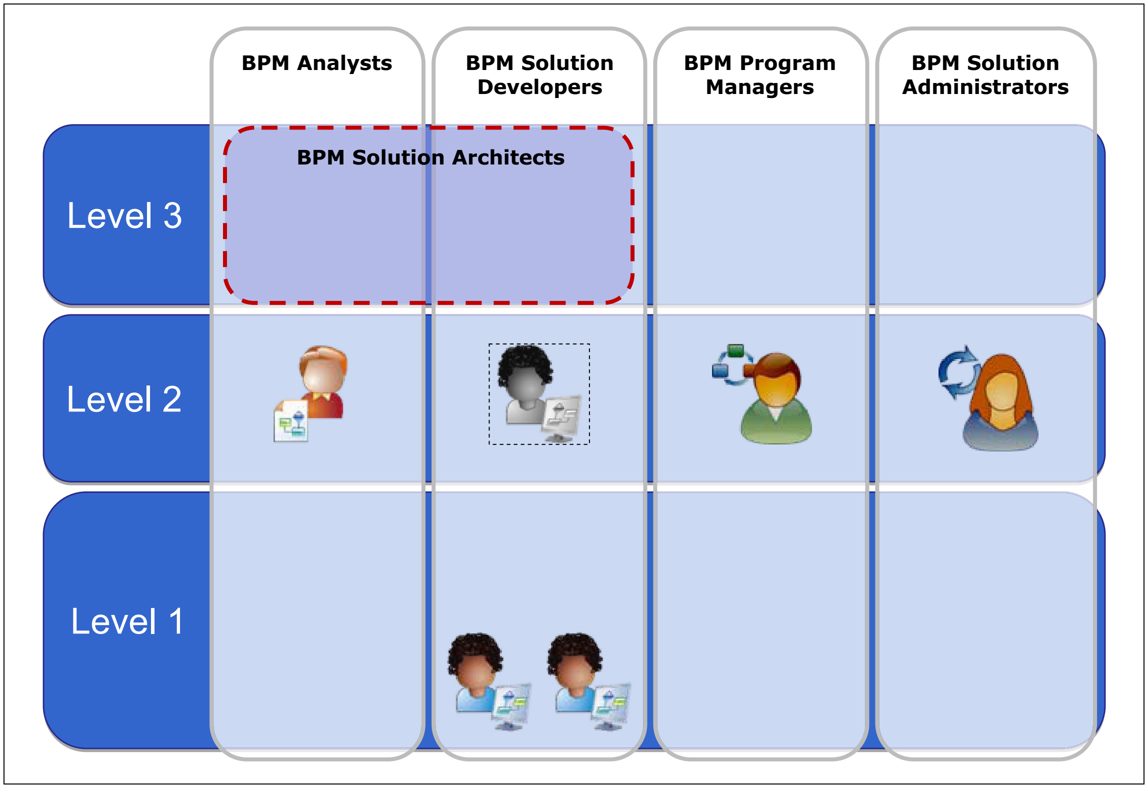


*Figure 4-5 This medium complexity process project scales the team's capability by adding more experienced individual contributors*

Again, the aim is to organize a team that can deliver the necessary scope in 90 - 120 day release cycles. There might be two or more 90-day release cycles for this project before a real production go-live event, but we can manage scope in incremental releases with this small and high-performing team. This example adds only one headcount to the number of individual contributors on the team, and increases the team capability by scaling up the experience of most roles to Level 2. We want to keep the team small and cross-functional to maintain high-performing and self-directed team structures. In this medium complexity process project, we might be dealing with a single top-level business process and 2 - 3 subprocesses for a total of 15 - 30 individual process steps. The user interface requirements might include several medium and high complexity user interfaces (Coaches) for several of the steps.

In this example, we added another Level 1 BPM developer to work under the leadership of a Level 2 developer. For a process that has more scope in system integrations than in user interfaces and process complexity, we could add a BPM solution administrator instead. Nonetheless, aim to keep the team no bigger than six individual contributors.

In this medium complexity process project, we can expect the BPM analyst to spend 4 - 6 weeks on a full-time basis leading up to Playback 1. If broken into two or more 90-day release cycles, the BPM analyst might re-engage for the first Playback cycle of each new release and participate part-time as needed thereafter. Here too, the BPM solution administrator might participate less than full-time to assist with complex integrations and deployment at the appropriate stages of each release cycle. The BPM developers and BPM program manager are assumed full-time throughout. Similar to low complexity projects, a Level 3 BPM solution architect, BPM analyst, or BPM program manager might participate on a limited part-time basis to provide guidance and mentoring throughout the project's release cycles.

### *Staffing a high complexity process project team*

Some processes just cannot be broken into multiple projects. They might contain many steps (60 or more) arranged in more than one top-level business process and 10 or more subprocesses. There might be large number of different user interfaces required or several high complexity user interfaces as a minimum requirement for Release 1. Some processes involve many system integrations that are required for Release One. Some processes involve many different participant groups (10 or more) requiring discovery workshops with several different groups of subject matter experts. For these types of high complexity projects, we again scale up the output of the team by adding more experienced Level 2 and Level 3 resources.

### Scaling up larger teams with multiple tracks

We might also need to scale up the total headcount beyond the recommended team size of six individual contributors. The team illustrated in Figure 4-6 has 11 individual contributors. With two BPM program managers, split this larger team into two tracks to maintain two small teams of five or six individuals. Distribute the senior team members equally among the junior members to maximize mentoring and learning. Our experience shows that two smaller teams far outperforms a single larger team with the same number of individual contributors. The BPM program manager must coordinate dependencies between the tracks, but multiple smaller teams (tracks) work faster and spend less time in meetings than single larger teams.

For a high complexity project, the roles that might be part-time for a small or medium complexity project easily scale up to full-time. These individuals distribute their time across the different smaller teams (tracks).

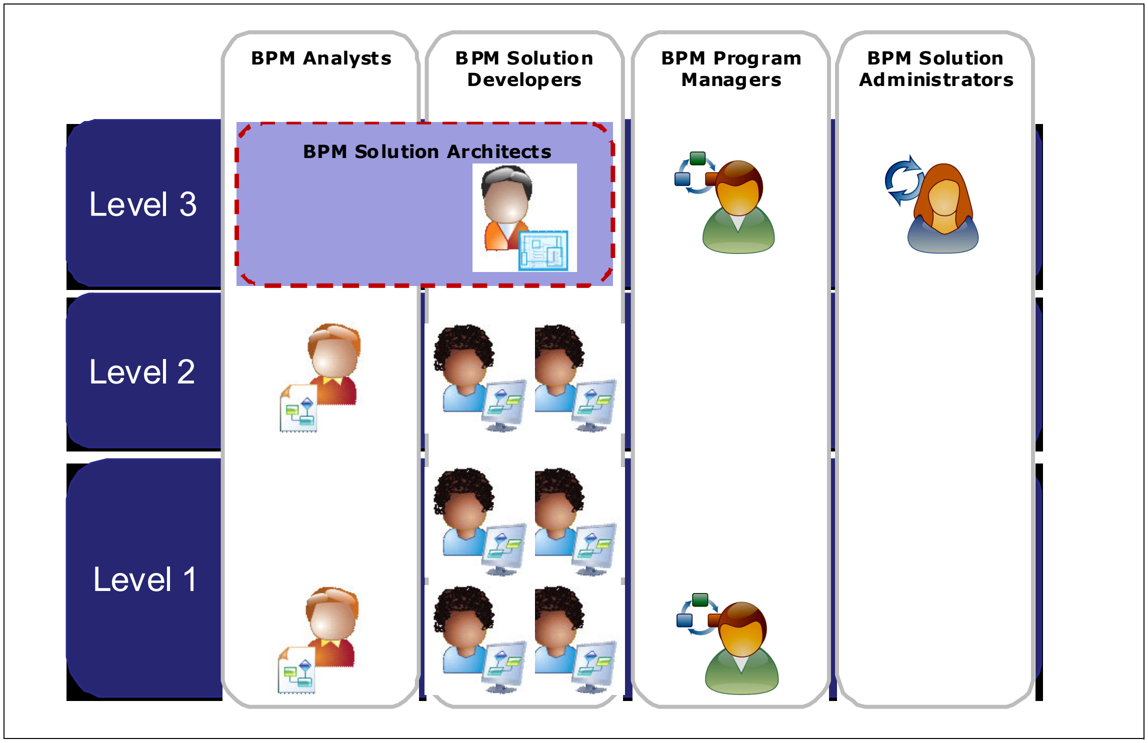


*Figure 4-6 Scale up for large complexity process projects by adding seniority and dividing a large team into multiple tracks*

### 4.1.4 BPM adoption starts with skills development

The journey to BPM transformation within your enterprise begins with skills development. Skills development leads to BPM adoption. BPM adoption leads to stakeholder acceptance, a new collaborative relationship between the operational business and the technology team, and additional funding for the BPM program and more BPM projects. Our experience is that organizations that are most successful in achieving BPM adoption make a conscious investment early and often in skills development activities, including training and continuous education, and formal and informal mentoring that accompany project-based experience.

In the next section, we review key concepts of agile methodology and tools that support BPM.

## 4.2 Agile planning and management for BPM projects

This section is not intended to be a reference of the merits of agile software development or a prescriptive manual of any specific agile method. There are several agile methods and here are a few you might recognize.

- Agile Modeling
- Agile Unified Process (AUP)
- Extreme Programming (XP)
- Feature Driven Development (FDD)
- Open Unified Process (OpenUP)
- Scrum
- Kanban (development)

There are other credible texts on these different methods. This section does, however, highlight key concepts of agile software development that are necessary for success in BPM. The Agile Software Development manifesto is worth repeating here in its short, simple, form:

> “We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:
>
> – Individuals and interactions over processes and tools
>
> – Working software over comprehensive documentation
>
> – Customer collaboration over contract negotiation
>
> – Responding to change over following a plan”[1]

[1] Source: http://agilemanifesto.org/

If BPM is a methodology that enables business agility, then agile software development is a methodology that enables software development agility. The operational business cannot be agile in measuring changes in business climate in real time and responding (managing) in short iterative cycles without a software delivery methodology that can also facilitate technology changes in short and iterative delivery cycles.

Agile software development depends on visibility. In the same manner that BPM requires visibility, an agile software development process requires constant real-time visibility into the state of development so that stakeholders can respond to constant change as assumptions are corrected and new information surfaces during development. Visibility into the current state of development includes some of the following items:

- How many hours of work remain in this iteration?
- What is the velocity (points) per iteration for this team?
- Are we on track to deliver the agreed upon user stories?
- What user stories are being developed during this iteration?
- Are we working on the highest priority user stories?
- What work is blocked and why?

### 4.2.1 Tools for planning and managing agile projects

In our experience, traditional project management tools like Microsoft Project, SharePoint, spreadsheets, and project wikis do not provide the level of granular visibility into the state of work (such as user stories, tasks, iterations, releases, defects, tests, and so on). They also do not provide the predictive planning necessary for successfully managing agile software development for BPM projects. There are a number of tools available for managing agile software development; some are more mature than others. For this book, we use IBM Rational® Team Concert™. Rational Team Concert is a segment leader for managing agile projects and artifacts. It can be downloaded from jazz.net community site (`https://jazz.net/`)and is available at no cost for up to 10 users with no time limit.

Rational Team Concert does require a server for installation and is accessible through a browser or an installable Eclipse based client. If server installation and management is an issue for you, there are many commercial tools available for managing agile projects in a SaaS/Cloud environment. As project data is stored in a cloud environment with these tools, you should review privacy issues and security concerns.

We chose to not endorse specific tools, but rather express the importance of the following items:

1. Select a tool that meets the real-time visibility demands of agile software development.
2. Complete adequate training on the tool for all participants.
3. Document and enforce working agreements so that information in the tool is current and useful.

### Non-software tools for managing agile software development

There are methods to manage agile software development projects that do not require any software at all. If the team is co-located, post-it notes that on a whiteboard are an effective method of tracking the backlog, the current iteration, and work that is complete. In this scenario, each post-it represents a user story with a point estimate and assumptions captured on the card. The whiteboard (or blank wall) has four columns: backlog, planning, in progress, and complete. This method makes reporting much more complicated, but in lieu of purchasing SaaS or obtaining a server to install Rational Team Concert on, it is a viable method.

No matter what agile software development tool you choose, success is directly related to whether the following tasks occur on a daily basis:

- Developers update their work (tasks, user stories, and defects) to reflect time spent, time remaining, and blocks (with reasons for the blocks).
- Project Managers address issues (blocks) and continuously monitor team velocity and team utilization adjusting the plan (iterations and releases) as team velocity changes.
- Stakeholders (business) collaborate with developers daily to resolve questions and remove blocks.

## 4.2.2 The transformational BPM journey

As a reader of this book, you are likely thinking about business transformation and building a BPM program. You should be envisioning and championing the idea of your company innovating and responding quickly to market changes, new developments, and improved efficiencies. You probably also have a laundry list of business processes that can be streamlined, made more efficient, and quality-improved. Achieving an organizational maturity where you can strategically think about the processes that are both core and ancillary to your business, along with implementing and improving them in a concerted effort, is the BPM journey.

## Thinking big, starting small, scaling fast

As with any journey, carefully choosing your path and building skills along the way is tantamount to successfully reaching your goals. Building a BPM program starts with a single successful project. It might not be your first project that champions agile delivery methods, and it might not be your second. But you do not scale to a BPM program until you achieve a successful delivery of a business process project with agile software development. Success should be measured in duration, business value, and reflection from project participants. A successful agile project becomes a proof point that your operational business can collaborate with the technology team to deliver business value in a short, iterative fashion. This first win garners stakeholder support essential for additional funding for new versions and new projects. It is important that your early project wins be chosen wisely to deliver business value (three to one ROI or better) and gain executive level visibility. A project that is too complex, or too political, could endanger the momentum of your BPM program. A project with too little visibility does not garner stakeholder support for the next project. Choose your projects wisely as you think big, start small, and scale fast.

## Scaling to a BPM program with successive project wins

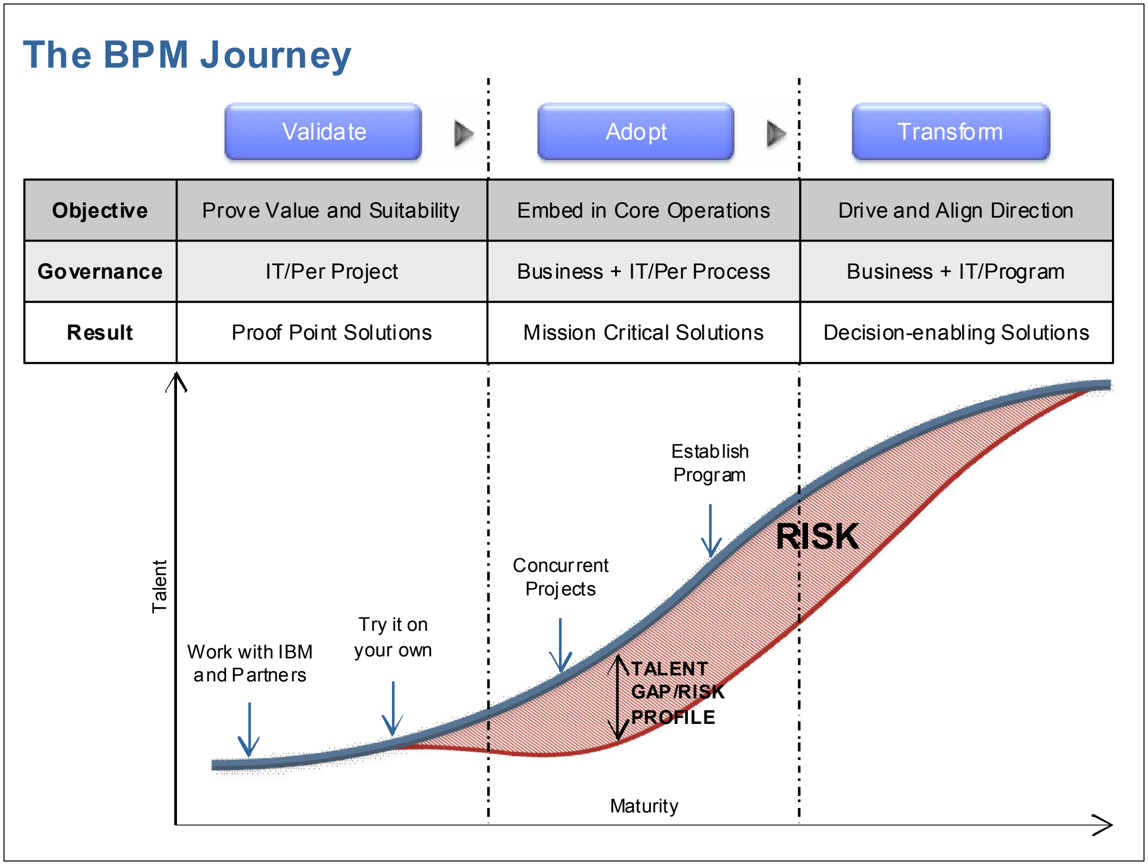
If your organization is new to agile software development and iterative methods of delivering technology projects, your first few agile software development wins should garner much attention. Stakeholders from both operational business and technology should build on the momentum of those early project successes. As you gain confidence with early project wins and learn from real project experience, your team tackles harder process problems with greater visibility and larger ROI.

Attempts to shortcut to complex projects with high visibility and significant ROI before learning tough lessons on lower complexity projects and achieving maturity in your teams' agile software development capability jeopardizes or postpones BPM program success. Achieving success on those first few low complexity projects is a necessary first step on your journey. You now have a pool of skilled individuals that can develop simple and medium complexity processes. You might be thinking about creating a BPM Center of Excellence (CoE) to organize teams, prioritize projects, distribute best practices, and manage shared process components and toolkits. Until these resources are truly mature, your organization's ability to drive and align business direction is limited to targeted processes.

## Entering the transformative period of BPM

Business transformation can be attained only by first validating BPM methods in your organization by achieving success on small projects. Following repeatable success on multiple concurrent projects, your organization begins rapid adoption of BPM. In the BPM program, success drives company-wide adoption. Your company begins a transformative period in BPM where the concept of prioritizing all work around business value, and achieving that value in short and iterative cycles lead by self-directed, cross-functional teams, becomes part of the fabric of your company.

Success with first projects creates a talent gap as skills development lags behind demand due to adoption (Figure 4-7). This talent gap must be managed by carefully selecting projects and increasing investment in skills development activities.



The BPM Journey

| | Validate | Adopt | Transform |
|---|---|---|---|
| **Objective** | Prove Value and Suitability | Embed in Core Operations | Drive and Align Direction |
| **Governance** | IT/Per Project | Business + IT/Per Process | Business + IT/Program |
| **Result** | Proof Point Solutions | Mission Critical Solutions | Decision-enabling Solutions |

*Figure 4-7 Talent gap*

### Validating agile delivery is a necessary early step in the BPM journey

The validation period ranges from six months to one and a half years, depending on the organization. A lengthy year-long project prolongs this validation period. It is better to plan multiple small projects that accelerate skills development and increases organizational confidence in agile delivery methods and tools. Delivering value quickly also helps gain momentum, excitement, and visibility within the company. One technique for delivering a process in short and iterative releases is to first build the high-level elements of the process that provide visibility, guide users with tasks, and capture key performance metrics. With subsequent releases, additional process steps (such as exceptions), more complex data capture, user interfaces, deeper tracking and reporting, and process automation of some steps can be added.

How do you pick the process to build for the validation phase? Section 3.2, “Identifying business processes with an inventory” on page 47 explains how to capture information associated with a process based on effort (size, complexity, and risk) and impact (pain and business value). It might be helpful to chart different projects and compare both the effort and impact.

Figure 4-8 is an example of how to visually represent a comparison of project candidates. Other aspects to consider when selecting those first projects include corporate politics, availability of knowledgeable resources, and the appetite for BPM on behalf of the process owner and stakeholders.

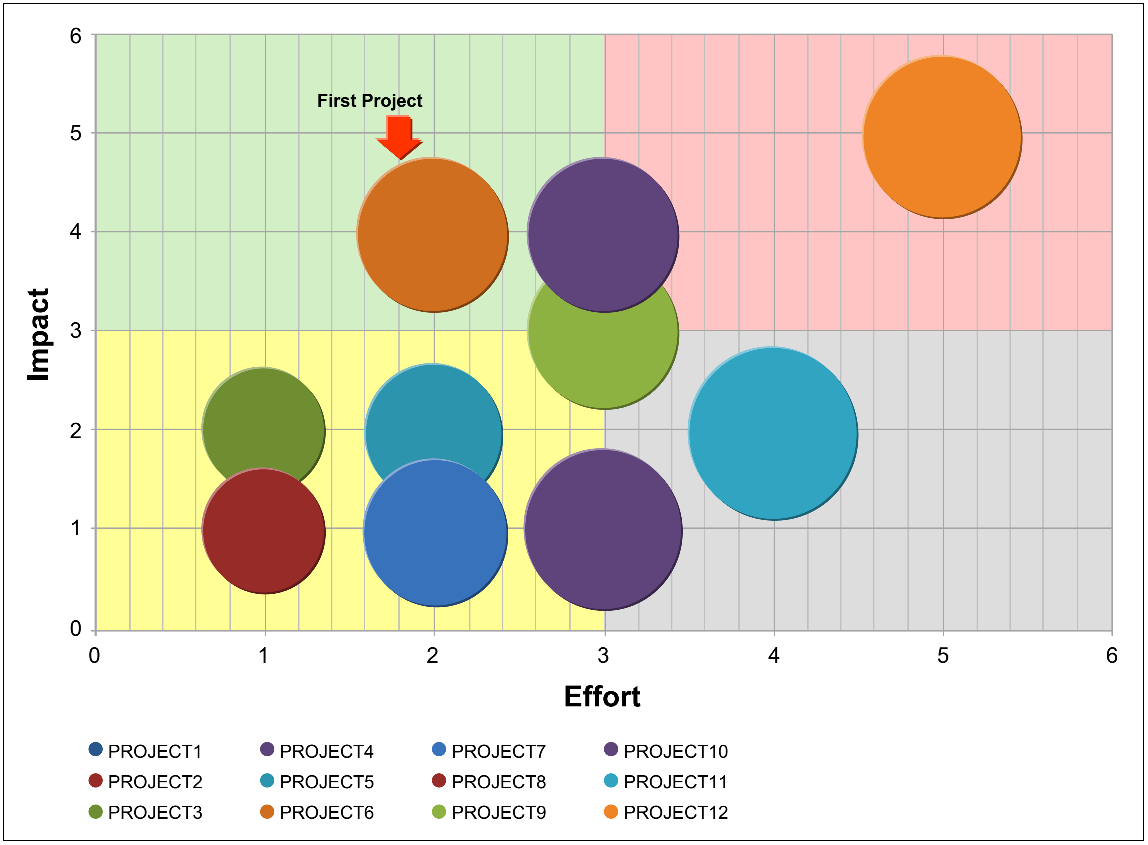


*Figure 4-8 Comparison of project candidates*

## 4.3 Estimating the BPM project scope

Most of this book explains the *value* of BPM. This section is about estimating the *cost* of BPM. If you are new to BPM and just starting your journey, you need to charter a project and ask for a budget. You might have a great story about business value and business impact, but your request for funding is incomplete without a ROI case that includes the cost and time associated with discovering, implementing, and deploying a business process with IBM Business Process Manager V7.5.

If you who have already started your BPM journey, you have experienced the success of at least one project. The attention that the project has garnered from stakeholders, executive management, and business users has generated a flood of new requests for BPM in other areas of your business. You need to estimate the scope of these incoming project requests as you build a process inventory. You need to assess each process before you can allocate resources for further process discovery and begin implementation.

### Estimating in Agile Software Development

An organization that has both experience and success in *Agile Software Development* techniques already has a different perspective on scope. Of the three variants in the *iron triangle* of project management (Figure 4-9), scope is the only one we *can* manage.

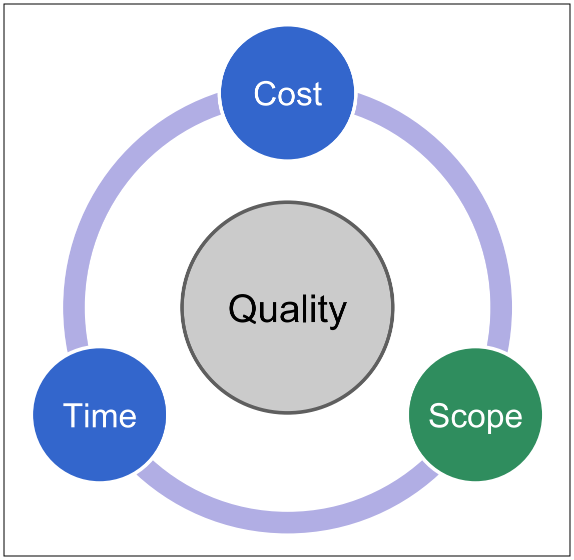


*Figure 4-9 Scope as a manageable variable*

Managing scope through better estimating techniques with transparency in both accuracy and precision better supports decisions that impact project cost and schedule. Essentially, we can manage project cost and schedule only by improving the way we estimate projects. We should concentrate on making our estimates more accurate and more precise, but *accept uncertainty*, and make decisions based on an appropriate level uncertainty as our knowledge of the process improves throughout the lifecycle of the project.

### New to agile software development

Few organizations have made a complete transformation to Agile Software Development. In BPM, having access to a business process management system (BPMS) does not mean that you are practicing BPM. Similar to BPM, using one of the many available tools or add-on packages for agile project management does not mean that your organization is agile. Like BPM, it takes time for your organization to adopt agile techniques, adjust to a method that works for your projects, and refine your method so that it is repeatable.

The organization that is new to agile software development, or has experimented with agile software development with limited success, relies on estimating methods that are more familiar. In the sections that follow, the *budgetary* estimate typically meets the needs of project stakeholders to charter a project, approve a budget, and staff resources. The budgetary estimate described in this book does, however, place an emphasis on quantifying and exposing uncertainty, which is a concept that is pivotal in agile software development.

### Unable to be agile

At the core of agile software development, there is *trust* between the customer (operational business) and the solution provider (internal IT department, vendor, Business Partner, and so on). It is trust between the customer and provider that allows both parties to start working on a project when time, cost, and scope are still uncertain. The customer agrees to participate in a collaborative fashion and pay for the project. The customer trusts the provider to work efficiently and produce *quality working software*. The solution provider trusts the customer to prioritize work and participate in short iterative delivery cycles (sprints or iterations) and accept the output in small chunks. Both customer and provider agree to prioritize around *business value*.

Sometimes the level of trust and collaboration required between customer and solution provider in agile software development cannot be achieved easily. Perhaps history and politics have strained the relationship between the operational business and the internal IT department. Sometimes contractual relationships, such as those with vendors or Business Partners, do not support agile software development. Often the capital expenditure process internal to your organization does not support funding projects (or large programs) without clear definition of scope up front.

These factors do not mean that agile software development *cannot* work in these environments. It means that it might take time to repair relationships and build trust. It means that vendors and partners need to constantly improve estimating with accuracy and precision and make it transparent when delivering estimates. It means that you need to accommodate our internal capital expenditure process but direct focus on the *business value* and impact within cost and scheduling constraints.

The different estimating techniques presented in this chapter help your organization estimate scope and calculate the cost of implementing BPM projects throughout your journey into BPM.

## 4.3.1 Prioritizing processes with high-level estimates

We estimate and refine the estimate throughout the lifecycle of a BPM process project (Figure 4-10).

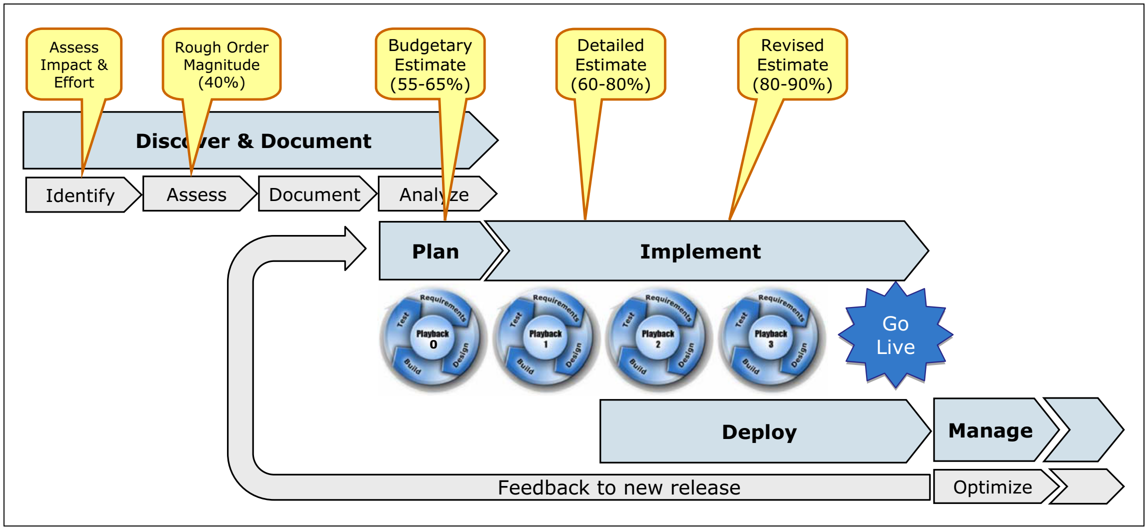


*Figure 4-10 Estimates provide necessary detail at decision points throughout process discovery, planning, and implementation*

In the beginning, during process inventory, we identify business processes and use a high-level assessment of business impact and implementation effort to prioritize the process for further discovery activities. Next we create a *Rough Order Magnitude* (ROM) estimate in as little as a few hours, but typically after 2 - 3 days in a discovery workshop. The time required to finish a ROM estimate depends on the complexity of the process and the availability of knowledgeable process experts. We use the ROM estimate to write a business case to justify another 1 - 3 weeks of discovery activities to include additional documentation and deeper analysis of As-Is and To-Be process models.

## 4.3.2 Planning for implementation requires accuracy and precision

As we move from process discovery to project planning and implementation, the additional time for documentation and analysis culminates with Playback 0. Playback 0 is accompanied by a *budgetary* estimate with improved precision and accuracy to a level of detail that we use to charter a project team and start planning for implementation. After process implementation (a project) starts, we use a refined *detailed* estimate after each iteration for planning future iterations, scheduling resources, and managing cost.

Use these different estimating techniques to provide project stakeholders an appropriate level of precision and accuracy to make educated decisions related to scope and budget throughout the project lifecycle from process inventory until process deployment.

### Using the cone of uncertainty to understand project estimation

The different estimating techniques presented in this chapter gradually improve in both accuracy and precision, thus reducing the cone of uncertainty as more process information is collected and understood during the project. The cone of uncertainty, as explained in *Software Project Survival Guide*, by McConnel and *Software Estimation: Demystifying the Black Art* by McConnel, is a core project management concept and central to agile software development. The key is to recognize that we can express an estimate only to a level of precision that correlates with the level of detail in the information available.

It is for this reason that, early in a process lifecycle, we can estimate only the process in terms of a low, medium, or high complexity ROM. As we learn more, we can better assess the scope in terms of *man-weeks* with a budgetary estimate based on projected process elements. When starting implementation, we further refine the estimate in *user story points* for planning iteration and release cadences. It might not be until a few iterations are completed that we can accurately estimate the remaining work in *man-hours* of development effort. It is imperative that decisions do not assume a level of accuracy and precision that is more specific than the supporting information used to create the ROM, budgetary, detailed, or revised estimates.

In Figure 4-11, we see how the relationship between the reported estimate (the curved blue line) gradually approaches the actual scope (1.0x) as the process matures from process identification to the playback milestones in process implementation.

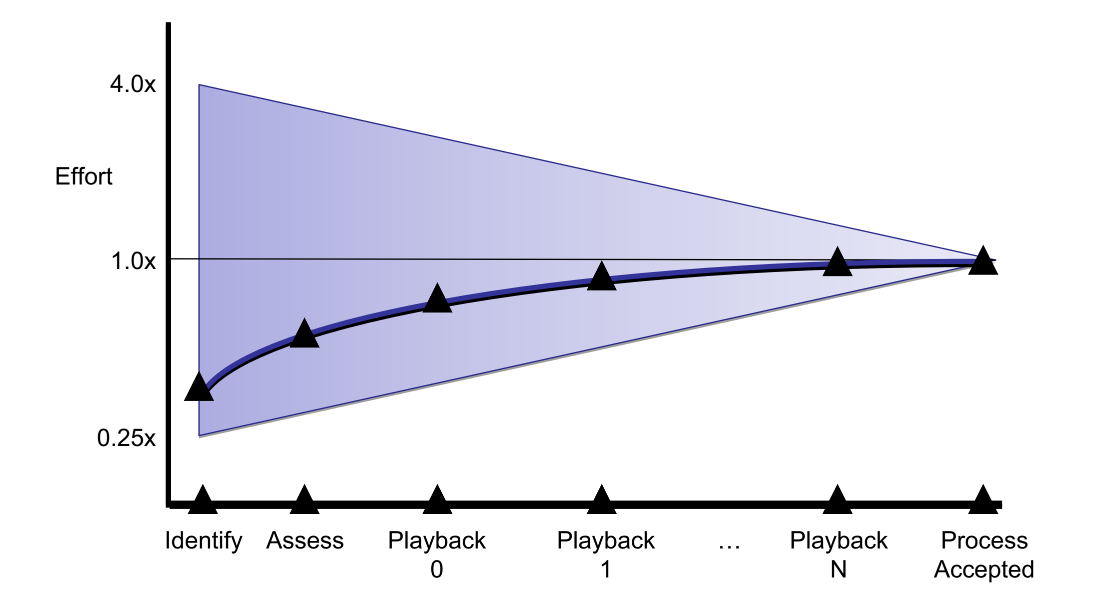


*Figure 4-11 The cone of uncertainty narrows as the estimate matures with added process knowledge and refinement in detail during implementation*

## Different estimating techniques for different decisions

The four different types of estimates used during the lifecycle of a process include the following types:

- Assess Business Impact and Overall Effort

  This estimate happens during process identification as you build your process inventory and is described in detail in Chapter 3, “Process discovery” on page 41.

- Rough Order Magnitude (ROM)

  A low precision and accuracy (40%) estimate used to build a business case for further process documentation and analysis. This estimate can be used in a business case to justify a project charter for process implementation.

- Procure Funds with Budgetary Estimate

  With added precision and accuracy (55 - 65%), this estimate is used for initial planning (cost, resources, and schedule) and is based on the outcome of process discovery and analysis.

- Plan Project with Detailed Estimate

  With refined precision and accuracy (60 - 80%), this estimate first appears during the first development iteration. This estimate is based on story points, used to bucket work to iterations, and commit assignments to developers, and should anticipate change as a percentage.

- Commit Schedule with Revised Estimate

  An updated detailed estimate based on actual story point velocity from multiple iterations. By Playback 3, the precision and accuracy is as high as 90%.

## Estimating with accuracy and precision expressed as uncertainty

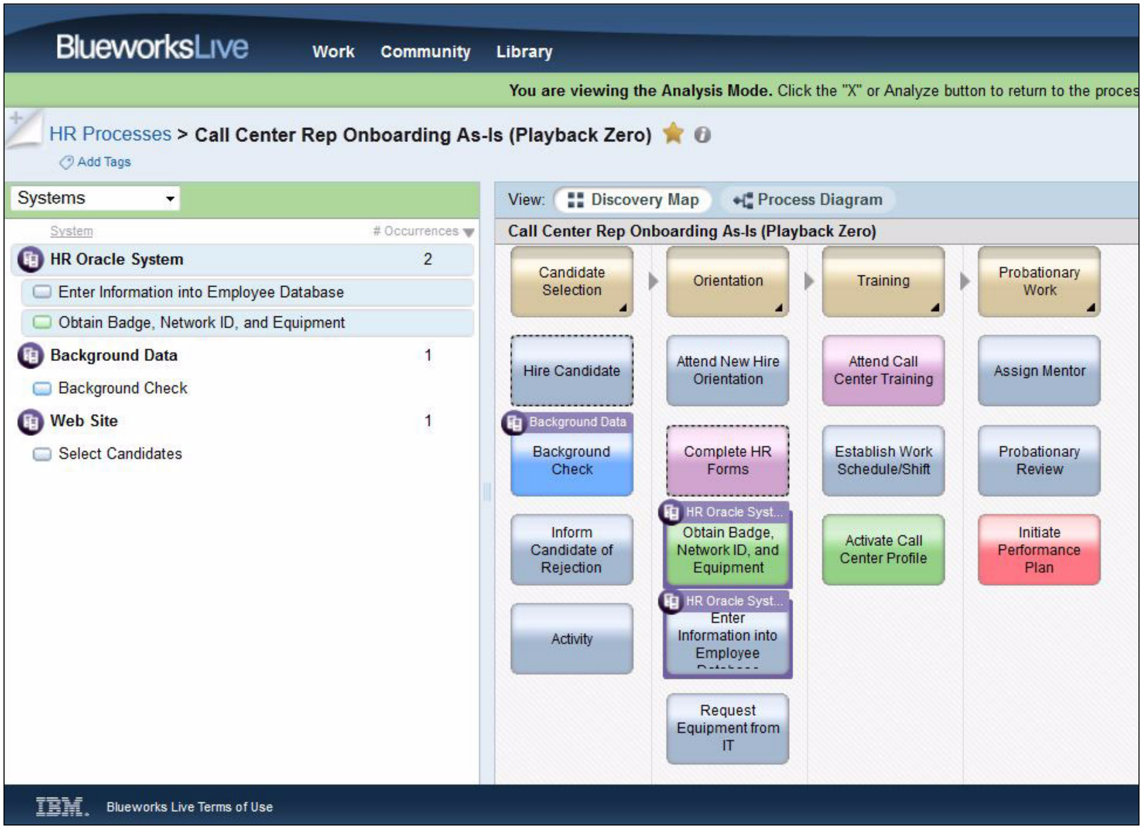
This section describes the latter three types of estimates in detail and how and when to use each method. As we explain each estimating method, it helps to recognize that estimates should be viewed as ranges rather than fixed values. The last several decades of research on software development suggest that early estimates range in an order of magnitude of four times. The actual scope could be as much as four times larger (4x) or four times smaller (0.25x) than our first estimate. In some environments, this range could be as large as 10x. Another important aspect of agile software development is to use historic data and retrospectives to make you a better estimator. Knowing your range of uncertainty when estimating different things at different stages of a project can help you make better decisions.

## 4.3.3 Using the rough order magnitude estimate

We can estimate a business process project as low, medium, or high complexity within an *order of magnitude*. This estimate can be used to build a business case for chartering a project initiative to further document and analyze the process. For some BPM teams, this estimate provides enough detail to support a business case and start implementation.

### Characterizing the process with high-level modeling

The ROM estimate requires a high-level process model of the As-Is process. We use an early version of the Blueworks Live process model to generate a quick count of attributes that characterize the complexity of a business process or process area (Figure 4-12).



**You are viewing the Analysis Mode.** Click the "X" or Analyze button to return to the proces

HR Processes > **Call Center Rep Onboarding As-Is (Playback Zero)**

| System | # Occurrences |
|---|---|
| **HR Oracle System** | 2 |
| Enter Information into Employee Database | |
| Obtain Badge, Network ID, and Equipment | |
| **Background Data** | 1 |
| Background Check | |
| **Web Site** | 1 |
| Select Candidates | |

**Call Center Rep Onboarding As-Is (Playback Zero)**

Candidate Selection
Orientation
Training
Probationary Work
Hire Candidate
Attend New Hire Orientation
Attend Call Center Training
Assign Mentor
Background Data
Background Check
Complete HR Forms
Establish Work Schedule/Shift
Probationary Review
Inform Candidate of Rejection
HR Oracle Syst...
Obtain Badge, Network ID, and Equipment
Activate Call Center Profile
Initiate Performance Plan
Activity
HR Oracle Syst...
Enter Information into Employee Database
Request Equipment from IT
IBM.
Blueworks Live Terms of Use

*Figure 4-12 Use the Analysis mode in Blueworks Live to gather and count processes, subprocesses, inputs/output, business entities, participants, and system integrations*

The effort required to briefly identify and assess the business process through process discovery can be completed in a 2 - 3 hour workshop with the right participants. For a larger, more complex process that requires interviews with several different SMEs, it could take 2 - 3 days of workshop time to generate the level of detail required for a ROM estimate.

The key elements of the business process model that we need for a ROM estimate include the following items:

- Top-level business process and subprocesses
- Participants (swim lanes) and number of steps/activities
- Unique Coaches (screens) across all human activities
- System integrations (that is, WSDL actions) used in human and system activities
- Business rules and policies (that is, complex activity routing or activity due date policy)

### Estimating scope and complexity by comparing historic process characteristics

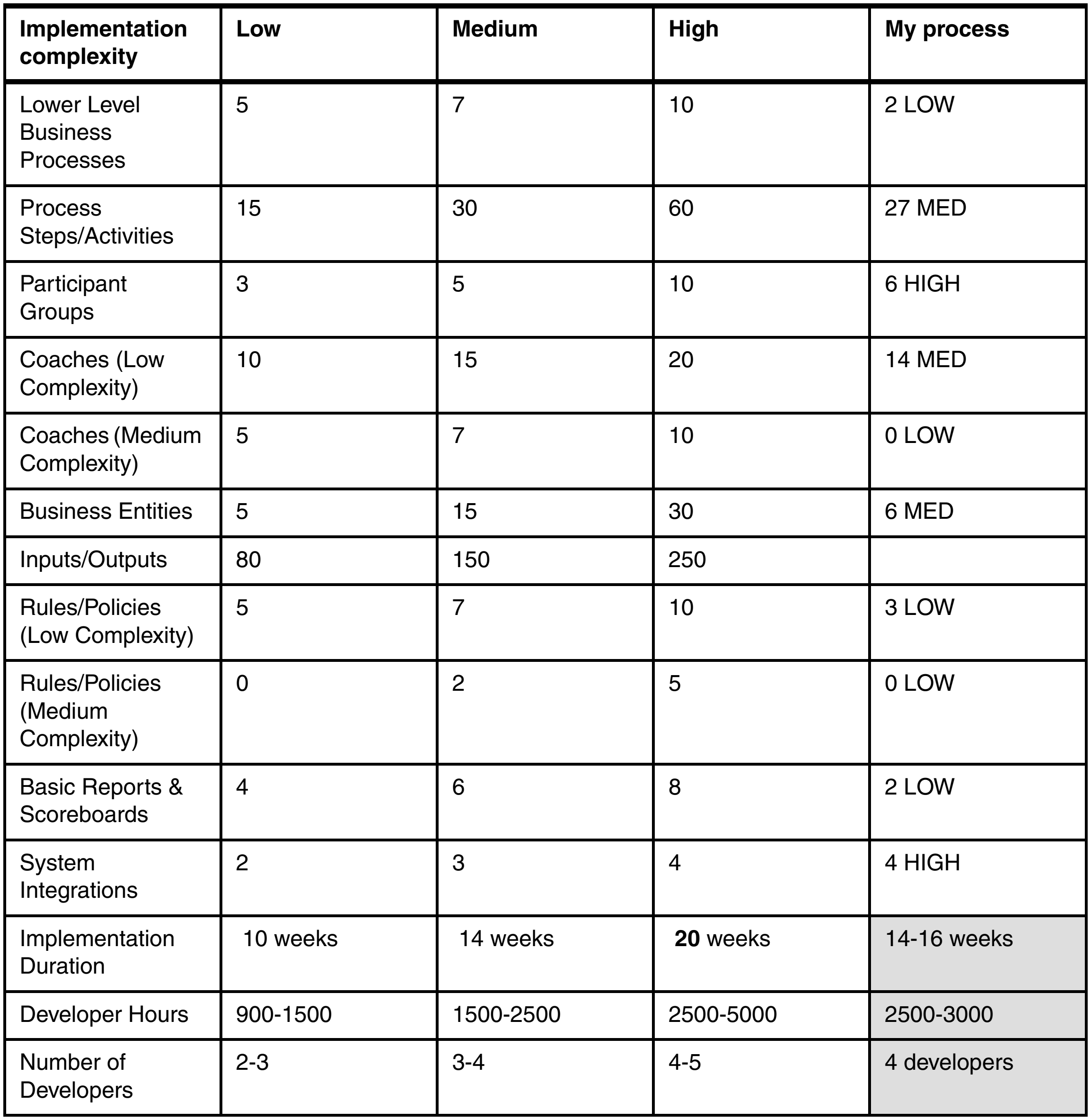
The ROM estimating method works by comparing measurable characteristics of your process to typical low, medium, and high complexity processes. Depending on how your process compares, we can characterize it as low, medium, or high complexity. Although the comparison of each metric is objective, the overall assessment of the size of your process is subjective.

Table 4-3 is based on data from hundreds of process projects aggregated by IBM over a period of nearly six years. This historic project data characterizes three deployment statistics in ranges to illustrate low, medium, and high complexity project profiles.

1. Project duration
2. Developer hours
3. Number of developers

*Table 4-3 A rough order magnitude estimate helps estimate a project size*

| **Implementation complexity** | **Low** | **Medium** | **High** | **My process** |
|---|---|---|---|---|
| Process Analysis | Yes | Yes | Yes | Yes |
| Top Level Business Processes | 1 | 1 | 2 | 1 LOW |



| Implementation complexity | Low | Medium | High | My process |
|---|---|---|---|---|
| Lower Level Business Processes | 5 | 7 | 10 | 2 LOW |
| Process Steps/Activities | 15 | 30 | 60 | 27 MED |
| Participant Groups | 3 | 5 | 10 | 6 HIGH |
| Coaches (Low Complexity) | 10 | 15 | 20 | 14 MED |
| Coaches (Medium Complexity) | 5 | 7 | 10 | 0 LOW |
| Business Entities | 5 | 15 | 30 | 6 MED |
| Inputs/Outputs | 80 | 150 | 250 | |
| Rules/Policies (Low Complexity) | 5 | 7 | 10 | 3 LOW |
| Rules/Policies (Medium Complexity) | 0 | 2 | 5 | 0 LOW |
| Basic Reports & Scoreboards | 4 | 6 | 8 | 2 LOW |
| System Integrations | 2 | 3 | 4 | 4 HIGH |
| Implementation Duration | 10 weeks | 14 weeks | **20** weeks | 14-16 weeks |
| Developer Hours | 900-1500 | 1500-2500 | 2500-5000 | 2500-3000 |
| Number of Developers | 2-3 | 3-4 | 4-5 | 4 developers |

The descriptive characteristics provided in Table 4-3 on page 137 serve as benchmarks for a candidate process to be implemented using IBM Business Process Manager V7.5. The table lists the characteristics associated with typical projects that range in size and complexity from low to high. Your project probably does not neatly fit into a single complexity column for all of the characteristics. These characteristics should be viewed as thresholds. If your project exceeds the count for a characteristic, assess your project by comparing it to the next level of complexity.

### Estimating a rough order magnitude for the Call Center Rep Onboarding process

The ROM estimate for the Call Center Rep Onboarding process used for the scenario in this book is illustrated in Table 4-3 on page 137. We counted the process characteristics and completed the last column in the table. For each characteristic, we compared the aggregated criteria and gave the candidate process a rank of low, medium, or high complexity. Out of the 12 criteria, we find two highs and three mediums. This process appears on the cusp of medium and high complexity projects (indicated with shaded cells in Table 4-3 on page 137). A ROM estimate for this process is at least as large as the maximum for a medium complexity project at 14 weeks for implementation with four developers and 2500 developer-hours. Because this process has high complexity attributes, the mid-point of a high complexity ROM defines an upper range at 17 weeks for implementation and 3750 developer hours. This estimate does not include cost or time for a BPM analyst, BPM program manager, testing, deployment, or SME participation.

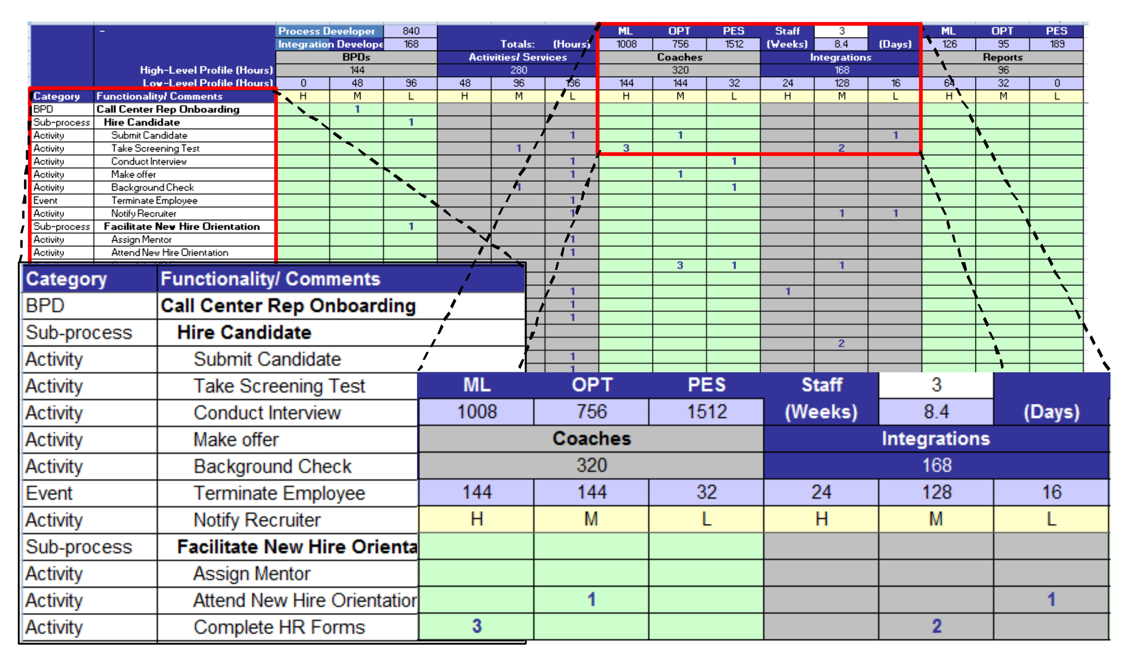
## 4.3.4 Planning a project with a budgetary estimate

Use a *budgetary* estimate when you need a higher level of confidence (reduced uncertainty) for project planning. In a budgetary estimate, the process requires a level of knowledge that can be achieved in a business process definition. The business process definition is modeled, documented, and analyzed to include human and system tasks, general user interface functionality, system integrations, key performance indicators, and reporting requirements. Although not all requirements are known at this time, you have enough detail to support assumptions about the level of complexity and scope to write an estimate that can be used to secure funding, plan a schedule, and allocate resources. The level of detail you need can be attained from the Blueworks Live process modeling artifacts documented and analyzed in process discovery during the weeks leading up to Playback 0. This type of estimate typically involves the BPM analyst or BPM solution architect working alongside a BPM program manager to complete a budgetary estimate template.

## Getting the budgetary estimate right

When we create a budgetary estimate too soon in the project without the necessary input, we risk causing more harm than good. The budgetary estimate requires a specific level of knowledge and detail about the process. Attempts to produce a budgetary estimate without the necessary input only results in making too many assumptions. These assumptions broaden and hide uncertainty. The real danger comes in presenting a budgetary estimate without making the uncertainty transparent. A budgetary estimate can also be dangerous if it is produced at the start of the project and is not refined after more knowledge is discovered and documented.

The budgetary estimate for the scenario in this book, based on using certified resources, has an estimate of pure construction work at 1008 hours for 9 weeks with three developers (Figure 4-13).



| | - | Process Developer | | 840 | | | | ML | OPT | PES | Staff | 3 | | ML | OPT | PES |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Integration Develope | | 168 | | Totals: | (Hours) | 1008 | 756 | 1512 | (Weeks) | 8.4 | (Days) | 126 | 95 | 189 |
| | | BPDs | | | Activities/ Services | | | Coaches | | | Integrations | | | Reports | | |
| | High-Level Profile (Hours) | 144 | | | 280 | | | 320 | | | 168 | | | 96 | | |
| | Low-Level Profile (Hours) | 0 | 48 | 96 | 48 | 96 | 136 | 144 | 144 | 32 | 24 | 128 | 16 | 64 | 32 | 0 |
| Category | Functionality/ Comments | H | M | L | H | M | L | H | M | L | H | M | L | H | M | L |
| BPD | Call Center Rep Onboarding | | 1 | | | | | | | | | | | | | |
| Sub-process | Hire Candidate | | | 1 | | | | | | | | | | | | |
| Activity | Submit Candidate | | | | | | 1 | | 1 | | | | 1 | | | |
| Activity | Take Screening Test | | | | | 1 | | 3 | | | | 2 | | | | |
| Activity | Conduct Interview | | | | | | 1 | | | 1 | | | | | | |
| Activity | Make offer | | | | | | 1 | | 1 | | | | | | | |
| Activity | Background Check | | | | | 1 | | | | 1 | | | | | | |
| Event | Terminate Employee | | | | | | 1 | | | | | | | | | |
| Activity | Notify Recruiter | | | | | | 1 | | | | | 1 | 1 | | | |
| Sub-process | Facilitate New Hire Orientation | | | 1 | | | | | | | | | | | | |
| Activity | Assign Mentor | | | | | | 1 | | | | | | | | | |
| Activity | Attend New Hire Orientation | | | | | | 1 | | | | | | | | | |
| | | | | | | | | | 3 | 1 | | 1 | | | | |
| | | | | | | | | | | | | | | | | |
| | | | | | | | 1 | | | | 1 | | | | | |
| | | | | | | | 1 | | | | | | | | | |
| | | | | | | | 1 | | | | | | | | | |
| | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | 2 | | | | |
| | | | | | | | 1 | | | | | | | | | |
| | | | | | | | 1 | | | | | | | | | |

| Category | Functionality/ Comments |
|---|---|
| BPD | Call Center Rep Onboarding |
| Sub-process | Hire Candidate |
| Activity | Submit Candidate |
| Activity | Take Screening Test |
| Activity | Conduct Interview |
| Activity | Make offer |
| Activity | Background Check |
| Event | Terminate Employee |
| Activity | Notify Recruiter |
| Sub-process | Facilitate New Hire Orienta |
| Activity | Assign Mentor |
| Activity | Attend New Hire Orientatio |
| Activity | Complete HR Forms |

| ML | OPT | PES | Staff | 3 | |
|---|---|---|---|---|---|
| 1008 | 756 | 1512 | (Weeks) | 8.4 | (Days) |
| Coaches | | | Integrations | | |
| 320 | | | 168 | | |
| 144 | 144 | 32 | 24 | 128 | 16 |
| H | M | L | H | M | L |
| | | | | | |
| | | | | | |
| | 1 | | | | 1 |
| 3 | | | | 2 | |

*Figure 4-13 Budgetary estimate for the scenario in this book*

This figure does not include meetings/consultation sessions, shared process development work (like building of toolkits or supportive/re-usable frameworks), or lead oversight. It calculates developer productivity at 40 hours per week. These other elements should be quantified based on experience. If you do not want to break down each of these elements, setting other activities and change at around 50% of the core construction value and planning developers at 80% productive results in a likely implementation of 12 weeks with four developers. Although it is ideal for the budgetary estimate to be less than the ROM, often what is discovered is more than what was initially set for the ROM.

### Increasing confidence with additional depth and detail

Like the ROM estimate, a budgetary estimate starts by taking an inventory of solution characteristics that include the following solution elements:

- Business Process Definitions (BPDs)
- System & Human Activities (process steps)
- Coaches (screens)
- System Integrations
- Key Performance Indicators (KPIs) & Service Level Agreements (SLAs)
- Scoreboards & Reports

Unlike the ROM estimate, this technique applies additional knowledge related to complexity and scope of each of these elements. 4.3.6, “Sizing solution components for processes” on page 155 reviews each of these solution elements in detail to illustrate how each is measured on a scale of low, medium, and high complexity, which can give us a cost range expressed in man-days (8 hour increments).

### Accounting for shared resources and activities

The budgetary estimate also includes accounting for the cost and schedule of those aspects of the project not directly accounted for in the solution artifacts. Some of these activities might span multiple projects and be managed at the program level, but must be paid for by project budgets.

The share resources and activities include:

- BPM analyst time for process analysis
- BPM solution administrator time for deployment planning and deployment activities
- BPM solution administrator time for environment topology design and installation of IBM Business Process Manager V7.5
- BPM program manager time for project management
- BPM developer time for troubleshooting and prototyping integration to new systems

- Team time planning and delivering Playbacks
- SME time for iteration planning, commitment, and acceptance
- SME time for user acceptance testing

### Applying pessimistic and optimistic multipliers to handle uncertainty in planning

Real-life process implementation never executes exactly as planned because of uncertainty in the original plan. Uncertainty comes from ambiguity in subjective estimates prone to human error. Uncertainty also stems from the variability arising from unexpected events or risks.

When you create a budgetary estimate to charter a new project and attain funding, it is important to recognize the uncertainty in the estimate. You accomplish this task using *optimistic* and *pessimistic* multipliers to give you a cost range in which the project most likely falls. These multipliers allow you to plan a project budget, schedule, and staffing plan with contingency recommendations.

The concept of applying optimistic and pessimistic multipliers is not new in agile software development or BPM. This technique can be traced back to Program Evaluation and Review Technique (PERT) analysis and Critical Path Method (CPM) more than 60 years ago. Historic data on estimating projects suggests that you are more likely to underestimate than overestimate. An optimistic multiplier ranges between 0.5 and 0.75, and a pessimistic multiplier ranges between 1.5 and 3. Use multipliers that match your confidence in the amount of uncertainty. If your budgetary estimate is 1500 hours and you are confident that you have a fair understanding of the level complexity and uncertainty, use an optimistic multiplier of 0.75 and a pessimistic multiplier of 1.5. Our budgetary estimate gives a 55 - 65% confidence interval that implementation falls between 1125 hours (0.75 x 1500) and 2250 hours (1.5 x 1500).

**Calculating an estimate:** With statistical analysis, you can use the standard deviation of a PERT distribution to calculate an estimate for planning purposes. The PERT distribution uses most likely, optimistic, and pessimistic values. A PERT+n estimate for calculating length of effort is calculated as follows:

PERT+n = average + n standard deviation

Where:

- Average = ((4 x most likely) + optimistic + pessimistic)/6
- Standard deviation = (pessimistic - optimistic)/6

Using an example project with rounded values from 1100 and 2300 hours of implementation, you get the following values:

- Average = ((4 x 1500) + 1100 + 2300)/6 = 1567
- Standard deviation = (2300 - 1100)/6 = 200
- PERT+0 = 1567 + 0 x 200 = 1567
- PERT+1 = 1567 + 1 x 200 = 1767
- PERT+2 = 1567 + 2 x 200 = 1967

A PERT+0 estimate is close to the most likely estimate and is appropriate for planning a project without contingency (50% probability based on normalized data). The PERT+1 value is used for planning out a project with contingency (68% probability of falling within one standard deviation). The PERT+2 value is used for budgetary procurement (90% probability of falling within two standard deviations) and assumes change as part of the effort.

You should not necessarily propose funding and schedule around the pessimistic implementation estimate (2300 hours in the previous example). You might seek a budget for a rounded PERT+2 value (such as 2000 hours). You then prepare to actively manage the scope to keep the project within schedule and cost constraints around 1800 hours, saving 200 hours for additional implementation costs and schedule contingency. You can expand the budgetary estimate to use this same model for estimating elements mentioned around shared resources and their activities to come up with an all encompassing budgetary estimate. How you use the budgetary estimate still largely depends on the nature of the relationship between the customer (operational business) and solution provider (internal IT department, vendor, partner, and so on).

### 4.3.5 Managing a project with a detailed estimate

Planning project releases and iterations with *story points* rather than hours comes from Agile Software Development. The Agile Software Development manifesto suggests that we value *customer collaboration over contract negotiation*. The trouble with using hours to plan happens when we present an estimate of 40 hours to a process owner. We set an expectation, or contract, that the functionality is delivered in the course of a week. Story points, however, give us a more arbitrary means to measure scope for planning a release or iteration with relative sizing of the user stories. This situation allows us to collaborate on a plan without confusing hourly estimates as schedule contracts with far more accuracy and precision than
we have.

**Remember:** Plan with points, manage with hours.

When we start managing project implementation with a *detailed* estimate, we use user stories and story points to size delivery components. Our detailed estimate should be revised at the end of every development iteration. As the project progresses, and the cone of uncertainty narrows, we can apply a narrower ranges of uncertainty. Using the earlier described PERT approach, we might restrict our revised estimate from PERT+2 to PERT+1, and ultimately move to PERT+0 in later iterations.

As we drill down in the granularity of what we are estimating, we have more detailed knowledge of solution artifacts and scope (Figure 4-14). Our estimates improve in both accuracy and precision. We use different measures at different levels of granularity.

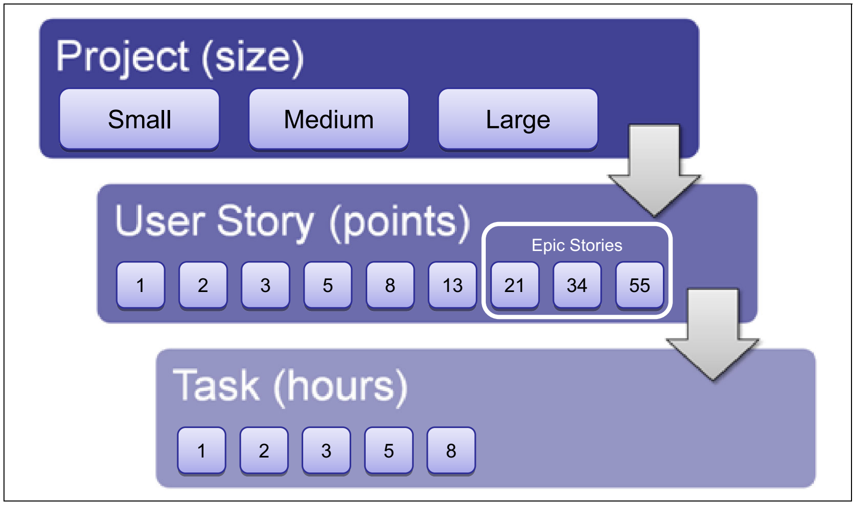


*Figure 4-14 Drilling down in granularity*

### Measuring scope with user story points

A BPM program manager collaborates with the development team when estimating user stories. If this estimating activity is performed by the lead developer alone, then the accuracy of the estimate is highly dependent on that individual's experience and whether they are generally optimistic or pessimistic. It is better to involve either the entire delivery team or a good cross section of the team. This situation allows developers to challenge each other and raise questions that others might not consider. User story point estimates based on general consensus are more accurate.

As the development team reaches consensus, they think about the various solution components the user story requires. Design considerations and questions are documented as notes or comments on the user story to provide justification for the estimate and provide useful information to developers during implementation.

Teams use *story points* differently, but the two most popular ways to measure using story points are either on a scale of 1 (low effort) to 10 (high effort) or a Fibonacci sequence (0, 1, 2, 3, 5, 8, 13, 21, 34, 55). The Fibonacci sequence is useful in that it reflects the manner in which uncertainty (vagueness) plays a part as the story gets bigger and more complicated. Both methods have pros and cons; what is important is that you chose a method that works for you and your team knowing that you can modify later if it is not working. When estimating story points, the team should take three concepts into consideration:

- Level of effort
- Level of complexity
- Level of uncertainty

### *Breaking down epic user stories*

User stories that cannot be implemented in a single iteration should be further broken down into smaller user stories, or substories. An example could be stories that have data integration dependencies or complex navigation. Some of these large stories have multiple facets that are specific to a complex feature; these stories are often referred to as *epic* user stories. Epic user stories are too large and contain too much ambiguity to estimate accurately. If using the Fibonacci sequence method of estimating story points, any story larger than 21 story points might be an epic. Over time, your teams recognize epic stories. Epic stories do not need to be broken down immediately upon discovering them. They need to be broken down only when loading an iteration and scheduling user stories for implementation.

### *Teaming with user story points*

The concepts a team uses to estimate a user story in story points (effort, complexity, and uncertainty) are subjective measures. The story points estimated by one team differ from the estimate by another team for the same user story. This situation is perfectly normal and should be expected. Each team of developers bring their own experiences and unique perspectives on what is complex. Team variations in prior knowledge and experience account for different estimates. If we assign user stories to a different team, the point estimates for the user stories should be revisited by the new team.

### *Measuring team velocity with user story points*

User story points are relative in the sense that two user stories that have the same story point value should be relatively the same size. This situation does not mean that two stories of the same point value should consume the same amount of development hours. We can have two user stories of the same estimate, but one could be complicated and have low effort while the other could be high effort but have low complexity.

The user story points help us measure scope so that we can plan releases and iterations. After completing several iterations with the same team, we have a good measure of the team's velocity in story points. This measure is key to planning the scope achievable in future iterations and releases.

**Hours worked in a day**: Do not assume developers complete 8 hours of development tasks each day. These estimates are for actual development effort and do not include time spent presenting mini-playbacks to process owners, collaborating with SMEs, building prototypes, attending meetings, creating documentation, and so on. Research suggests that experienced developers perform an average of 6.5 hours of development work each day. Developers with less experience, user stories with more ambiguity, and tasks involving high complexity further reduce the hours of work completed in a day.

## Writing user stories with Blueworks Live

During the discovery phase, the BPM analyst helps SMEs write their own user stories. Each activity in the business process contains at least one user story, but more likely has five or more user stories. User stories are a simple reminder that should prompt additional discussion and collaboration between the user story owner (SME) and the BPM developers during process implementation. Although Blueworks Live is not an agile project management tool, it is the first place we might capture user stories during process discovery, documentation, and analysis.

Before agile software development, traditional requirements were documented by analysts as functional specifications, or as a software requirement specification (SRS). The SRS included a logical breakdown of the component parts of the solution and use cases. User stories differ from traditional specification documentation, as they focus purely on the business requirement and do not take solution design consideration into account. Each user story should be no more than 2 - 5 sentences and should follow this format:

*As* a participant, *I want* to do something *so that* I achieve a business objective.

The most important part of the user story is the *so that* part that illustrates the *business value i*n the user story. This business value helps all members of the team stay focused on delivering that business value, and less concerned with the systems, technology, design patterns, and solution detail we might use to implement the user story. The business value is also important in prioritizing user stories for project planning.

Capturing requirements in user stories has a number of benefits:

- User stories allow the analyst to focus on the requirement and defer design considerations to a later stage, thus speeding up analysis.
- User stories can be prioritized and sized independently.
- A user story can be allocated to a specific iteration during development.
- User stories can be allocated to individual developers for implementation.
- User stories can be used to manage change control.

### *Finding the user story in activity documentation*

A business process definition might contain several levels of subprocess before reaching activities that are performed by a human or system. The activity description should describe the steps that are performed by the participant. For example, the Take Screening Test activity has the following activity description:

> The candidate completes a 30-minute test so that the HR Manager can assess the candidate's proficiency in languages (English, Spanish, and Chinese). The candidate also completes a 5 minute typing test so the HR Manager can rate the speed and accuracy of their typing skills.

As illustrated next, the documentation often contains several user stories. These stories can be written in the documentation and captured in Blueworks Live for all project participants to review and comment on.

**User story**: As the HR Manager, I want the candidate to complete a 30-minute test so that I can assess their proficiency in languages.

**User story**: As the HR Manager, I want the candidate to complete a 5 minute typing test so that I can rate the speed and accuracy of their typing skills.

The BPM developer is responsible for implementing every aspect of a user story. For a user story, the solution might require several artifacts that span the following architectural layers of a process application:

- Business Process
- Activity, Coach
- Business Object Model
- Integration Services.

Sometimes a user story cannot be implemented in its entirety in a single iteration when all of these layers are considered. Furthermore, dependency on integration development outside the process application can block completion of the integration components. In these cases, a user story can be subdivided into multiple substories to implement and delivery Process, Activity, Coach, and Integration level components independently.

### *Avoid writing passive user stories*

These user stories might seem reasonable. However, these stories are examples of *passive* user stories. The main participant (candidate) is less significant than the beneficiary (HR manager). It is better to write active user stories where the main participant is the individual that is performing the activity. This situation is how users are expected to interact with the BPM solution. When requirements are expressed as active user stories, you might discovery additional user stories, as shown in the following shaded box.

**User story**: As the Candidate, I want to complete a 30-minute test so that the HR Manager can assess my proficiency in languages.

**User story**: As the Candidate, I want to complete a 5 minute typing test so that the HR Manager can rate the speed and accuracy of my typing skills.

**User story**: As the HR Manager, I want to review the results of the candidate's screening tests so that I can eliminate candidates that do not have the appropriate skills.

Throughout discovery and implementation, we continue to identify missing or new user stories and add them to the backlog. User stories provide a set of requirement *reminders* for the BPM solution. While writing user stories, SMEs should also consider the following types of user stories:

- Ad hoc activities
- Administration
- Measurements, Process Visibility, and Reports
- Non-functional requirements

Taking these user stories into consideration, the user stories shown in the following shaded box were captured for the *Take Screening Test* activity in our example

**User story**: As the HR Manager, I want to gather screening test metrics so that I can assess the quality of candidates that recruiters are putting forward.

**User story**: As the HR Manager, I want the ability to define different screening tests so that the screening process can be used for various positions.

### *Managing the user story backlog*

The project backlog is the queue or container for all user stories not yet completed or scheduled for an iteration. At the start of the project, all user stories are entered into the backlog. As we plan new iterations, we pull user stories out of the backlog. As new user stories are created during the project, we add them to the backlog. At the start of each new iteration, the process owner can prioritize brand new user stories alongside previously written user stories. In this method, the process owner has control over the scope of the project. It is typical that the backlog increases in size during the first release of the project. This situation means that the team, including the process owner, SMEs, and developers, all help manage scope by creating user stories (or reminders) of additional capability identified during the project. The project should end with user stories in the backlog ready for planning another release.

### *Prioritizing user stories*

The process owner and SMEs are responsible for prioritizing user stories. This priority helps the process owner work with BPM developers during iteration planning to identify which stories to add to the iteration and work on first. When prioritizing the backlog, it helps to simplify the view and arrange stories into three groups: high, medium, and low. Give each user story a priority indicating the business value the user story represents in correlation with corporate and project objectives (Table 4-4).

*Table 4-4 Assigning priorities to user stories*

| Priority | Assessment | Description |
|---|---|---|
| High | Must have | I cannot do my job without this item. |
| Medium | Have a workaround | It is painful, but I can do my job without this item through a workaround if I know that it is temporary. |
| Low | Nice to have, or Not required | This feature would make my job a lot easier, but it is not in the critical path.<br>This feature is not necessary or pertinent for me to do my job. |

Here are some best practice for prioritizing user stories.

- If arranging into three groups, aim to distribute evenly: Top third, middle third, and bottom third. *It is not helpful if more than 50% of all user stories are High priority.*
- Prioritize user stories without regard to size. Try not to let the size of a story artificially influence priority.
- For Epic stories that are medium or high priority, break them up into substories and prioritize each one.
- If the user story does not have clear and transparent business value, or business impact, ask the process owner and user story owner (SME) to finish writing the story before setting the priority.

### Planning, committing, and accepting iterations with user story points

The process owner collaborates with the development team to plan iteration themes and load iterations with user stories. We use story points to measure scope so that we can schedule resources that align with the duration of the iteration. After loading an iteration with user stories, BPM developers review the user stories and define work items (development tasks) for each user story. This activity is collaborative, as developers rely on the user story owners to answer questions, provide examples, and elaborate on the details needed to start implementation.

> **Estimating scope in hours:** For the first time in the lifecycle of the project, we have enough detail to estimate the scope in hours.

After all user stories have development tasks estimated in hours, the team collaborates again to verify that the estimated hours fit in to the scheduled duration of the iteration and resource plan. If the iteration is overloaded, process owners remove user stories based on priority until the iteration fits the resource schedule. At this time, the process owner and development team *commit* the iteration. Upon completion of all development tasks for a user story anytime before the iteration ends, the process owner *accepts* the user story as completed. After all the user stories are accepted, the iteration is *accepted*.

## Estimating development tasks with hours

The developers (and only developers) are responsible for estimating the hours needed to complete each development task. A best practice is that no development task should be more than 8 hours. Some teams prefer to use a Fibonacci sequence (0, 1, 2, 3, 5, 8) for tasks. Tasks larger than 8 hours should be further broken down into multiple tasks, or subtasks. Tasks larger than 8 hours leave too much room for uncertainty due to ambiguity or variability due to unplanned events or risks.

Another best practice is for developers to update the estimated hours remaining on their tasks daily. If the work remaining increases due to new knowledge or unforeseen events, the risk is immediately visible in the iteration burn-down chart and the team can react accordingly.

## Managing development tasks in iterations

Iterations, also referred to as *sprints*, are the time-boxed containers for managing development. Iterations are important in establishing project cadence, setting delivery expectations with the process owner, and measuring the progress of work completed. Here are some best practices for managing development with iterations:

- Iterations should be *planned* 3 - 5 days before iteration starts to give the team time to define and estimate development tasks.
- Iterations should be *committed* before the scheduled iteration start.
- Iterations should be *accepted* before the scheduled iteration end.
- Iterations should be scheduled to maintain the same duration 1 - 3 weeks in length.
- Teams new to agile software development should plan shorter iterations (1 - 2 weeks).
- Experienced teams can plan longer iterations (2 - 3 weeks), but never more than 4 weeks.

If new information about development work surfaces during the iteration, or assumptions change, user stories can be removed from the committed iteration. However, no *new user stories* should ever be added to the iteration after the iteration is committed and started. This situation does not mean that developers cannot start work on other user stories. It does means that a *committed* iteration is an agreement between developers and process owner and it is unfair to add scope to an iteration after it is committed and started.

### *Monitoring iteration burn-down*

After an iteration starts, we monitor burn-down of effort remaining in hours and burn-up in user story points accepted (Figure 4-15). This situation means that the team takes hours and reset estimates for effort remaining as development tasks are updated daily. This situation also means the user story points accepted should trend upward as user stories are accepted by the process owner.

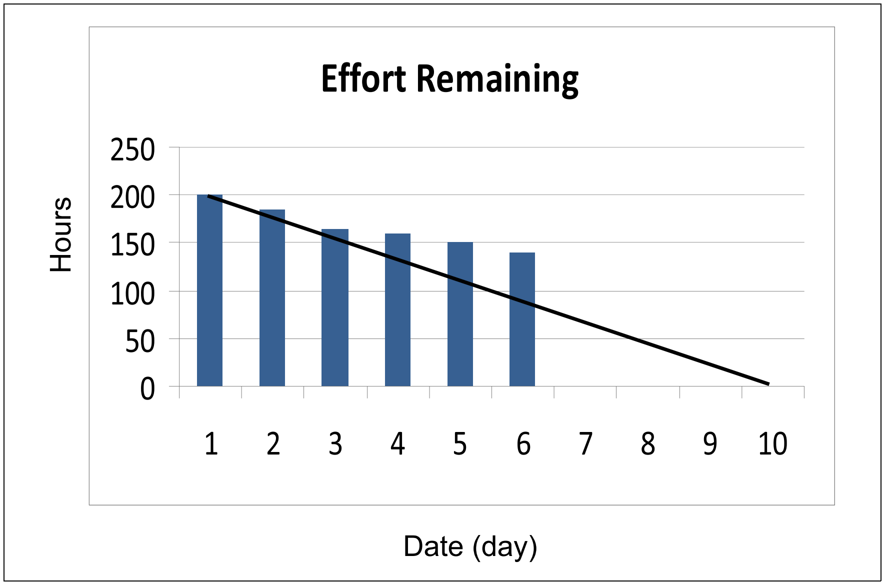


*Figure 4-15 Example iteration burn-down chart on day six of a 10 day sprint with a trend line*

The goal is to accept all user stories before the end of the iteration. This goal is not always possible and sometimes a user story either requires more work than originally estimated, or the user story is not accepted by the process owner. If the story requires more development effort before it can be accepted, we can remove the user story from the iteration and schedule it in the next iteration. The team gets no credit for user story points in the current iteration, as no *working software* is accepted by the process owner.

In our example scenario, we plan two-week iterations (10 workdays) for a team of three developers at various levels of experience. As there are not quite eight full hours in a workday, we plan iterations with no more than 200 hours of development tasks (three developers x 10 days x 6.5 hours per day). The example iteration burn-down chart in Figure 4-16 shows an iteration in trouble. The work remaining is not trending downward at a rate that completes all the work by the end of the iteration.

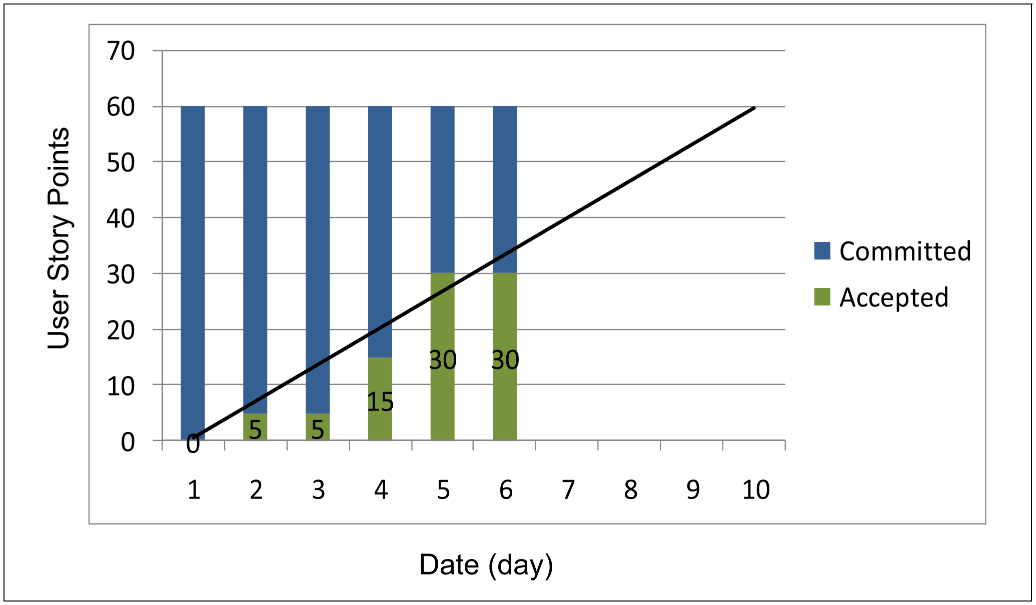


*Figure 4-16 Example of iteration burn-up in user story points*

Figure 4-16 shows the user story burn-up for the same iteration on day six. The iteration was planned with 60 user story points. Days two through four are below the trend line, but the iteration appears to be back on track by day five with 30 user story points accepted.

### *Do not correlate story points with hours*

It is common for teams new to agile software development to try to correlate story points with hours. Resist this temptation! Creating a correlation between points and hours (for example, 5 hours per point) invalidates the benefits of planning in points. It also assumes homogeneous experience and expertise.

If you want to associate time with points, then the time should be done as ranges for each point. You should use a model that incorporates uncertainty, such as a Fibonacci sequence (for example, five points would relate to a range of 32 - 48 hours). The range should be tighter for smaller points (for example, 6 - 10 hours for a single point estimated story) and greater for larger point estimates (for example, 40 - 80 hours for an eight-point estimated story).

#### *Using team velocity instead*

Our desire to correlate story points with hours is based on our motivation to forecast and plan. We ask ourselves, "how long does it take to implement 100 story points?" We can answer this question with *team velocity*. Team velocity is measured in story points accepted per iteration and should be based on historic team performance to include the most recent 4 - 6 iterations (teams improve velocity over time). We can apply a range to our estimates to include optimistic, most likely, and pessimistic values when planning releases or iterations. If we do not have historic team performance data, we use an initial estimated team velocity to plan a series of iterations based on various sized teams. After the team delivers several iterations, we use actual team velocity and further refine the project schedule.

### 4.3.6 Sizing solution components for processes

In any estimating method, we rely on baseline assumptions. In the following sections, our estimates assume the following items:

- All time-based estimates assume that the work is completed by resources at a *Level 1* proficiency, as described in 4.1, "Achieving BPM maturity through skills development" on page 100.
- The term *development* includes time for collaborative design, solution implementation, and unit testing.
- All estimates assume that IBM Business Process Manager V7.5 is used to its fullest capability for all solution components, unless specific exceptions are called out and sized individually.
- Estimates assume that process documentation and analysis is completed to a level of detail equivalent with Playback 0 or Playback 1. This situation means that the Business Process Diagrams including the details (SIPOC analysis) for each step are mostly complete and placeholders can be estimated for the areas requiring additional documentation.

The following sections provide ranges for the level of effort required given characteristic levels of complexity. When applying these ranges, steer to the higher end of the range when dealing with high levels of uncertainty.

### Business Process Definition

The characteristics that affect the level of effort to construct a Business Process Definition (BPD) are as follows:

- Number of activities (milestones, subprocesses, and tasks)
- Number of decisions (splits, joins, and gateways)
- Number of events (start/end, timer, message, and exception)
- Number of participant groups (swim lanes)
- Number of user stories documented in the activities

BPD development work includes the creation of BPD artifacts in Process Designer and all of the configuration of the swim lanes, milestones, events, and activities in each BPD. This work does not include the development work to *implement* the activities or events with human services, decision services, or other service development in Process Designer. The range in development effort (measured in man-days) required for BPDs of low, medium, and high complexity is illustrated in Table 4-5.

*Table 4-5 Range in development effort*

| BPD complexity | Steps/Activities | Decisions/Events | Participant groups | Estimate |
|---|---|---|---|---|
| Low | < 10 | < 2 | < 4 | 2 - 4 |
| Medium | 10 - 15 | 2 - 5 | 4 - 8 | 3 - 6 |
| High | 15+ | 5+ | 8+ | 8+ |

When estimating the overall complexity of a BPD, you need some level of detail as to the underlying complexity. Table 4-5 characterizes BPD complexity as a number of steps, decisions, events, and participant groups. The number of steps includes process milestones, subprocesses, and tasks. Decisions and events include any decision gateway or split on the process diagram. Events include start/end events, message events, timer events, and exception events. It helps to assess the relationship between events, gateways, and steps in the process model to get an idea of the number of valid pathways through the process. Is there a clear *happy path* and few exception paths? Or are there several dozen paths and loops in the process diagram? The latter case might have a high degree of uncertainty or much process complexity and should be reflected in your estimate.

## Coaches and Services

Coaches are the user interface windows that allow a human user to participate in a process. Coaches show process data (simple and complex variables) to users and accept input from the user. Users can enter data fields into coaches or act by clicking buttons or other web form controls. There are a number of factors that influence the complexity of a coach, including the relative number of data elements and number of actions a user can take on a coach (Table 4-6). Form field validation on the coach also influences the level of effort required to finish implementing a coach.

*Table 4-6 Factors influencing the complexity of a coach*

| Activity/Service/ Coach | Number of data elements | Percentage requiring validation | Buttons/User Actions | Estimate for each |
|---|---|---|---|---|
| Low | 10 - 15 | 30% | OK, Cancel | 4 - 16 hours |
| Medium | 20 - 30 | 50% | 5 | 16 - 32 hours |
| Complex | 50+ | 75% | 10 | 32 - 64 hours |

For a non-coach service, there are a series of considerations. They are shown in Table 4-7.

*Table 4-7 Considerations for non-coach services*

| Service type | Typical estimate for each service |
|---|---|
| Task Service | 6 - 12 hours |
| Data Access Service | 6 - 12 hours |
| Utility Service | 6 - 12 hours |
| Event Service | 4 - 8 hours |
| WebService Wrapper Service | 8 - 24 hours |
| Integration Wrapper Service | 4 - 8 hours |
| Initialization Service | 4 - 8 hours |
| Action Service | 4 - 8 hours |

## Scoreboards and Reports

To properly estimate the level of effort required for sizing a single scoreboard, information needs to be defined. Specifically:

- Does a report require multiple subqueries to produce?
- With what frequency do reports need to be refreshed in the UI?

- Is there any need for static reports?
- Are there chart types or visualizations that we do not know how to produce?
- Is the data being captured/readily available?

Collecting information to answer these and similar questions help you break down the complexity of reports into the categories shown in Table 4-8.

*Table 4-8 Report categories*

| Report | Number of cross-referenced factors | Number of filters | Number of EPVs | Number of Tracking Points required | Number of drill-down levels | Estimate for each |
|---|---|---|---|---|---|---|
| Low | 2 | 1 | 1 | 1 - 10 | 0 | 12 - 24 hours |
| Medium | 3 - 4 | 2 - 4 | 2 - 4 | 11 - 20 | 1 | 24 - 48 hours |
| Complex | 5+ | 5+ | 5+ | 20+ | 2+ | 48 - 80 hours |

Table 4-8 does not include ancillary functionality such as:

- Multiple reports per scoreboard
- Filtering and drill-downs
- Events/Alerts based upon scoreboard results

### Data persistence and business data system of record

It often happens that data persistence and creation of a business data system of record (BSOR) is overlooked during early project estimates. Whether a BSOR is needed might not be known until more details are known. This topic is a good one to discuss during process discovery and should be accounted for in a budgetary estimate. Here are some questions that can help during the early phases of a project:

- Is there an existing (legacy) BSOR?

  For many processes, there already is a system (or multiple systems) that are the “source of truth” for business data. Examples of business data might include customer records, invoices, and purchase orders. In our sample scenario, employee records are maintained in an “HR Database” that serves as the BSOR. For some processes, business data has no formal (or governed) system of record. The business data might exist in email, Excel files, or a shared drive. Asking a few questions early can help expose what might be a significant impact to the budgetary estimate.

- For each business entity identified in process discovery, where is that information stored today?
- If there is a BSOR, do other systems integrate with the BSOR today? If yes, how?
- Has there been mention of needing a “custom database” for business data?
- If we need to create a BSOR, do other systems need access to this data?

The order of magnitude per BSOR is shown in Table 4-9.

*Table 4-9 Order of magnitude per BSOR*

| Complexity | Approach | Estimate |
|---|---|---|
| Low | ► No requirement for a new BSOR.<br>► Only the process needs access to business data.<br>► Primarily to support reporting. Could use Performance Data Warehouse. | 0 - 5 days |
| Medium | ► A BSOR must be created.<br>► Only the process needs access to business data.<br>► DBA involvement required.<br>► The database must be normalized and follow client standards. | 5 - 20 days |
| Complex | ► A BSOR must be created.<br>► Process *and* external applications must access business data.<br>► DBA involvement required.<br>► Database must be normalized and follow client standards.<br>► EAI/SOA. | 20+ days |

### System Integrations

Implementing a BPM solution typically requires integration with other systems within the enterprise (including LDAP, ERP, or other operational data stores). BPM provides a framework and tools that make these integrations as painless as possible.

In general, for integrations, use the following criteria to determine the amount of work:

- The number of systems to integrate.
- The manner of the integration.
- The number of integrations per system.

In general, we plan 1 day per integration point and an additional 5 - 15 days per system (providing that direct access can be provided and no prerequisites need to be worked). Factors that affect this type of effort include the following ones:

- The first integration to any system can double the days to get it done. We call this process the “Handshaking integration”.
- Does the integration exist in IBM Business Process Manager? (If yes, this integration lowers the estimate.)
- Does the integration exist for another system? (If yes, this integration lowers the estimate.)
- What is the number and complexity of the data elements? (The more elements and complex, the higher the estimate.)
- Can we realize efficiency? (The 100th integration of the same type should take less time than the first.)
- If the integration team is using IBM Business Process Manager to unit test their integration time increases.
- This time does not include the creation of the integration on the external system.
- Java API integrations typically take 50% more time, and more time for ones dealing with binary data (document management, for example).

## Other considerations

Other elements that should be considered when sizing and scoping a project include the following elements:

- Regulatory Compliance
- BPM Governance
- Policy regarding testing and change control
- Infrastructure Considerations
- Customizations to the BPM platform to support the deployment
- Notifications/Escalations
- Non-Process Deliverables: Functionality not in the process tasks, such as maintenance screens and search capabilities

5

# Implementing a BPM project

This chapter describes the methods and tools used in implementing a BPM project with IBM Business Process Manager. For detailed technical instruction about how to implement a business process, visit the IBM Business Process Manager Information Center or the IBM Business Process Manager Community wiki, where you can find several “cookbooks” focused on specific aspects of process implementation.

We begin with playback concepts and its best practices. *Playback* is the methodology that we adopted during iterative business process development. The first playback (Playback 0) is a key activity during the discovery phase.

Next, we introduce the authoring environments provided with IBM Business Process Manager V7.5. The remaining sections describe you how you can practice the playback methodology by implementing the Call Center Rep Onboarding scenario in IBM Business Process Manager V7.5. Finally, best practices are introduced.

# 5.1 Business process implementation overview

Business process implementation is a critical step in the lifecycle of Business Process Management (BPM). In this phase, business processes are implemented iteratively, from a business idea to a practical and executable business process application.

It is important to keep the *business* part of BPM in focus during the implementation phase. Often, the development team reverts to more traditional development styles, focusing a disproportionate effort on technical details. This situation is a common failure point for BPM projects in which a constant vision for business value and ownership is required. As described in 4.1.1, “Cultivating skill sets in different roles” on page 100, business roles are heavily involved in the discovery and documenting phases. Now is *not* the time to end their involvement.

Many of the benefits associated with successful BPM projects, such as reduced time to market, realized business value, and shortened test cycles, rely on an iterative development lifecycle to achieve them. For the implementation phase to fulfill the iterative methodology, we use a model called playbacks.

A *playback* is a focused demonstration of a partially implemented process application, delivered to the business and IT stakeholders for discussion, consensus building, and approval. Playbacks give stakeholders the opportunities to provide feedback that drives the next iterations of process development.

IBM Business Process Manager allows rapid playbacks of the process definition at any point in the development lifecycle. In the Process Designer tool, you are developing a model for discussion and visualization, but also an actual model-driven, runtime solution. At any point in development, you can run the process, its subprocesses, and its services to validate your design.

## 5.1.1 Anatomy of a playback

A playback is not simply the execution of a portion of the solution or an ad hoc demonstration. It is an opportunity to involve the participants of the process in a concrete and valuable way. The business users should run the playback, with coaching where needed. Each playback provides validation that the process is headed in the correct direction and fosters business ownership, expectance, and sponsorship of the solution. The overall benefit of this exchange cannot be overstated.

Figure 5-1 shows the role of playback in the solution delivery cycle.

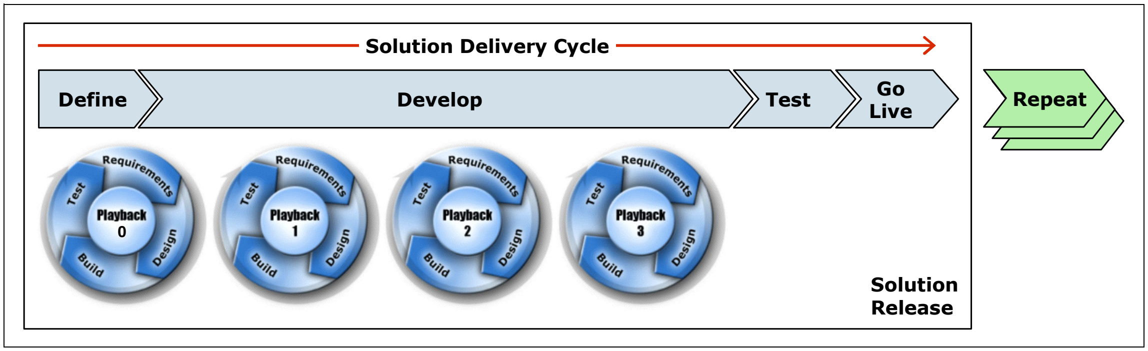


*Figure 5-1 Solution delivery cycle*

Each BPM project should have at least four *milestone playbacks* with the following audience:

- All business participants (at least one representative of each group)
- All business stakeholders in the process
- Report consumers (decision makers)
- Process owners
- Process developers
- Engagement manager
- Sales team (at least for initial playbacks)

Each playback should be presented with clearly stated goals and expectations, and an outline of where the playback boundaries fit within the overall process solution. The audience needs to be aware of what portion of the process they are focusing on. This situation also anchors conversations and questions on the elements of the playback instead of wandering off the path toward process discovery of other areas.

You must resist any attempts to derail a playback with lengthy conversations about interface changes, design or layout requirements, and items not pertinent to the functional, business-value portions of the process.

A playback within the iterative lifecycle is expected to create questions and suggestions that feed into subsequent playbacks. With each milestone reached and each playback exercise completed, the business participants play a stronger role in the development. They become more familiar with the solution and contribute to the outcome.

The intent is to execute the playbacks from the beginning in a production-like manner. Do not use the designer tools to run the playback. Instead, allow the business participant to use the default portal as they plan to in production. Start from the beginning each time and take note of the pain points and suggestions regarding human interactions.

To facilitate a well-organized implementation plan, develop a *playback strategy* that ensures that all playbacks fit together to effect a successful process implementation.

At a high level, a playback strategy consists of:

- A published schedule of playbacks that defines significant project milestones
- A comprehensive plan that covers all required elements of the solution
- Sign off after each milestone playback

A playback strategy that focuses solely on the technology runs the risk of taking the value proposition for granted. A process that does not deliver value is a failure:

- Demonstrate return on investment (ROI) using established metrics. If metrics have not been determined, it is of immediate priority to do so.
- Custom reports and scoreboards need to be tested and refined as much as the process itself. Do not wait until the last playback to show a report. Scoreboards should be designed before the process is modeled.
- Explain how the scoreboards and reports relate to the process metrics and facilitate management decisions.
- Practice the playback demonstration. Technical problems during a playback can undermine stakeholder confidence in the solution.

### 5.1.2 Business process development lifecycle

The business process development lifecycle in IBM Business Process Manager is a demonstration of the playback strategy. Having a good understanding of the business process development lifecycle is crucial in the business process implementation phase.

Figure 5-2 shows the lifecycle of a typical process development effort.

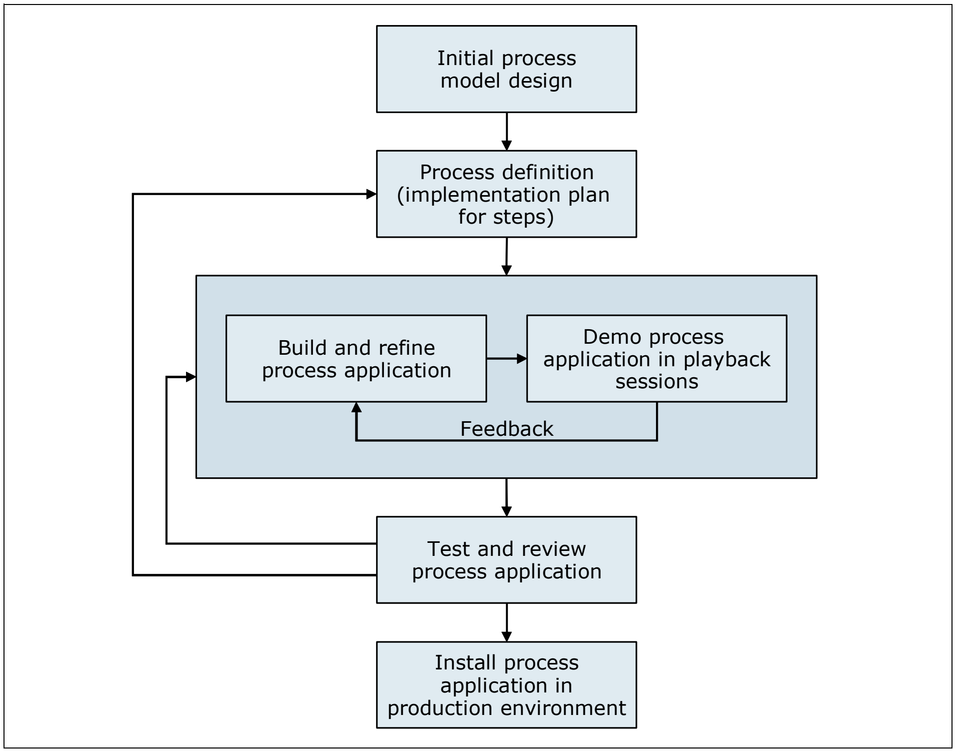

Initial process model design
Process definition (implementation plan for steps)
Build and refine process application
Demo process application in playback sessions
Feedback
Test and review process application
Install process application in production environment

*Figure 5-2 Business process development lifecycle in IBM Business Process Manager*

As you begin implementing a business process, you need to decide where to start your business process. You can start your business process implementation from scratch or based on an existing business process model that is formally described based on standards. Then define a high-level process based on the inputs from the business analyst. This high-level business process definition should be as easy to absorb as possible, by abstracting the details into higher-level elements where you can.

After the highest level business process is defined, you can start building and refining the business process based on the planned iterations. At the completion of or during each iteration, you can play back the business process to make sure that it functions correctly and conforms to the original business requirements. The results of the playbacks are passed back to process implementation where needed changes are made, and the next playback demonstrates those changes.

When the overall business process is implemented, you can test and review the process application. If any further changes are needed, go back to each stage of iterative development or even to the highest level process definition.

## 5.2 Implementing business processes using IBM Business Process Manager V7.5

IBM Business Process Manager 7V.5 provides a Process Center and authoring tools to support collaborative business process modeling and integration service development. Playback concepts are intrinsically supported in the IBM Business Process Manager V7.5 tools.

Figure 5-3 shows an overview of the implementation environment in IBM Business Process Manager V7.5.

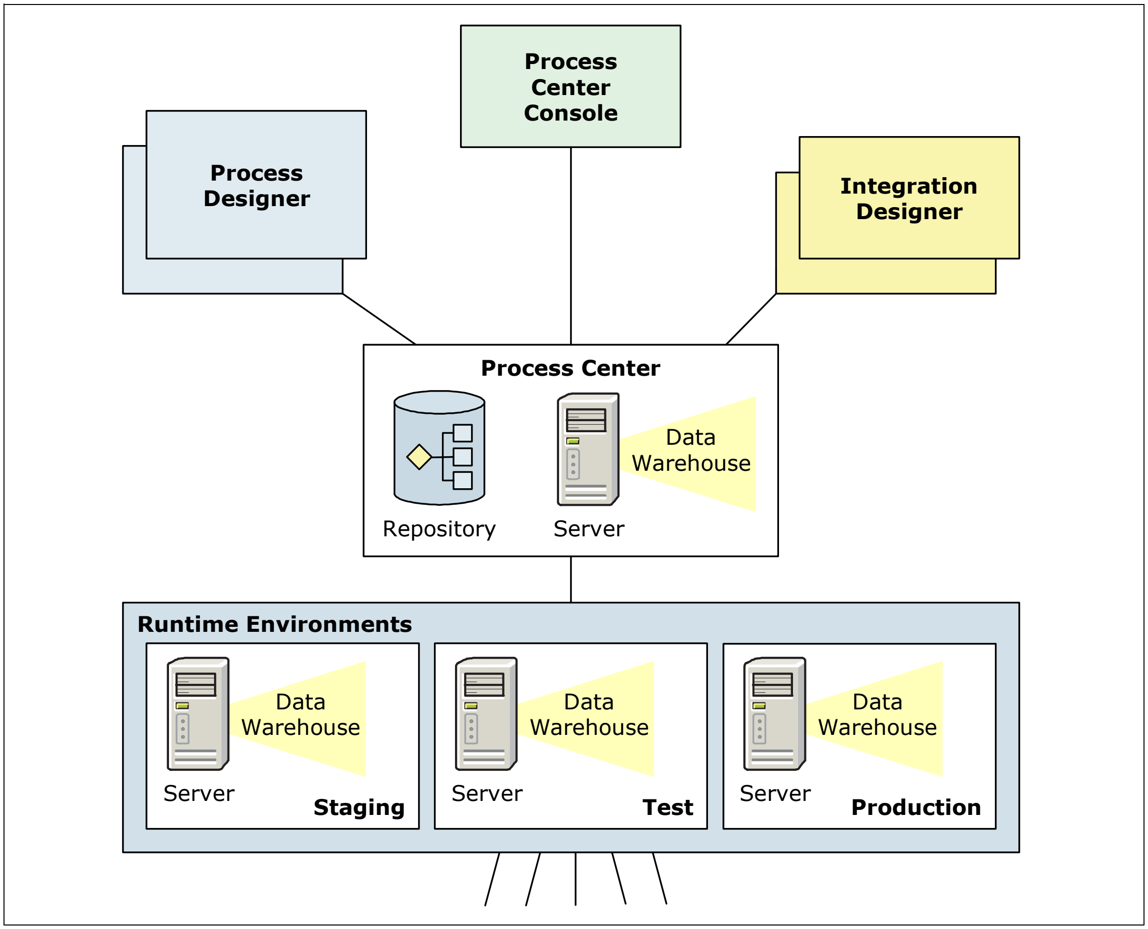


*Figure 5-3 Implementation environment*

## 5.2.1 Process Center

IBM Process Center serves as a central repository for all project assets that are created in Process Designer. When multiple Process Designer clients connect to the Process Center, users can share items, such as processes and services. Users can also see changes made by other users in real time as they happen. The Process Center can also be used as a repository for assets created in IBM Integration Designer.

Process Center includes a Process Center Server and a Performance Data Warehouse These features empower Process Designer users to run their process applications and store performance data for testing and playback purposes during development efforts, without having to deploy process applications to a separate runtime server.

From the Process Center Console, administrators manage running instances of process applications in all configured environments.

## 5.2.2 Process Designer

IBM Process Designer empowers you to model and implement your business processes and easily demonstrate process design and functionality during development efforts. Process Designer is where BPM developers design and implement business process definitions (BPDs.)

### Important concepts

The following concepts are important for BPM developers when implementing business processes in Process Designer:

- Process application

  A process application is a container for process models and their supporting artifacts. The process app and its artifacts are stored in the Process Center repository. After the artifacts are created and stored in the repository, they are assembled into a process application and a snapshot is taken. You can test, install, and administer the process application snapshot.

  Process applications contain some or all of the following artifacts:

  - One or more process models, also called business process definitions (BPDs)
  - The services required to implement activities or integrate with other systems
  - Service Component Architecture (SCA) modules
  - Toolkits
  - IBM Integration Designer libraries
  - One or more workspaces
  - An IBM Business Monitor model for monitoring business performance

- Service

  You can use services to implement the activities in a BPD. When a BPD starts and the steps (activities) are started, services can perform the required functions. The type of service that you choose to create depends upon the requirements of the activity. For example, if an activity requires integration with an external system, such as a database, you can create an integration service. (Process Designer provides many commonly used integration services in the System toolkit.) If an activity requires that call center personnel enter data about customer requests, you can create a human service with a coach.

### 5.2.3 Integration Designer

Often, a business process needs to call an external system such as Enterprise Service Bus (ESB), Enterprise Content Management, Extract Transform and Load (ETL) software, or to involve business processes across departments or enterprises. There can also be a need to start remote systems running on various platforms.

IBM Integration Designer empowers BPM developers to implement integrations to any external system necessary to complete the overall business process. It is available as a component in IBM Business Process Manager or as a stand-alone toolset for other uses. For more information, visit:

http://www.ibm.com/software/integration/business-process-manager/#

## 5.3 Methodology and design guidelines

This section describes how to apply the playback methodology and design guidelines to implement a business process, iteratively.

### 5.3.1 Playback planning

You need to have a playback plan before you begin implementation. A playback plan can apply to any business process implementation effort.

#### Playback 0

*Focus* on high-level business process understanding and building consensus.

This playback is done at least once at the beginning of a project as part of the process discovery phase.

The *goals* of a typical Playback 0 are:

- Consensus building and discovery of business process among stakeholders
- Further understanding of implementation scope
- Alignment of bottom-line expectations, KPIs, and high-value metrics from executive sponsors
- Transferring context and responsibility from the BPM Analyst to the BPM Developer

The *deliverables* typically include:

- An executable process definition (BPD)
  - Modeled to the level of depth necessary to show each discrete user task encountered in the process.
  - Does not need to include the specific implementation of each activity. A placeholder is sufficient.
- A participant and user group model

  As denoted by swim lanes in the BPD or as notes on the diagram when routing rules are more sophisticated than a single participant group
- Notes on the diagram to denote user activities that require information from external systems (integrations)
- A basic data model using IBM Business Process Manager 7.5 variable types

  Should cover all *process data* (as one subvariable type) and as much *business data* (as another subvariable type) as reasonably as possible to discover in the time frame
- Mocked-up reports that demonstrate a combination of the following BPM principles:
  - Visibility
  - Analysis
  - Control
- A focused demonstration of the previously mentioned deliverables, run within the default user portal interface, implemented by the BPM Analyst, and delivered by the process owner with coaching from a BPM Analyst

After conducting Playback 0, the next step in subsequent playbacks is for the BPM Analyst to act as the executive-level voice of the customer, through playback alignment with the executives.

### Playback 1

*Focus* on user interface design and implementation.

This playback is typically done at least once, and in concept it can be done as many times as is necessary to realize the theme of user interfaces.

The *goals* of a typical Playback 1 are:

- Consensus on and implementation of all necessary user interfaces
- Consensus on required data model to support user interfaces and business decisions

By working together with the BPM Analyst, the BPM Developer defines and implements each human interface that the process requires. This process should include all human tasks in the process, any ad hoc interfaces that exist outside the process, and any reports, dashboards, and scoreboards that are needed to elevate visibility and control of the business process.

The major *deliverables* of this playback typically include:

- A definition of the data model necessary for the business process
- A definition of what parts of the data model are captured through human tasks or interfaces
- A definition of the business actions that need to be enabled in each interface
- A definition of all possible next steps for each action
- A definition of the required validation necessary to maintain data and decision integrity
- The general appearance of the process solution (styles, themes, headers, consistent layout guidelines)
- An implementation of all required user interfaces as informed by the previous points

  It is assumed that this deliverable does not include the implemented integrations, reference data (auto-population of choices), or system-of-record population that would eventually be required for the full solution.
- A focused demonstration of the previously mentioned deliverables, run from the default user portal interface, implemented by the BPM Developer, and delivered by the process owner with coaching from the BPM Analyst and BPM Developer

The next steps after Playback 1 use the understanding of the process from Playback 0 with the data model and user interaction understanding from Playback 1 to focus on building the necessary integrations to support the process, and its decisions and human interactions.

### Playback 2

*Focus* on integrations.

This playback is typically done at least once, and in concept it can be done as many times as is necessary to realize the theme of integrations.

The *goals* of a typical Playback 2 include:

- Implementation and exception handling for all integrations that are needed to support the business process
- Definition and acceptance on all service level agreements and expectation settings with the owners of any external systems involved in the integrations

The BPM developer and technical consultant define and implement each integration necessary for the process. This process should include any external integrations and any System of Record (SOR) development necessary to support the full process solution.

The major *deliverables* of a typical Playback 2 include:

- Definition of the interfaces required for each integration point
- Definition of the data transformation required to send and receive information from these external systems
- Definition of all the fault codes that could possibly be returned from the external systems in response to starting an integration point
- Definition of the exception handling mechanism around handling any of the fault codes
- Definition of the required validation against these integration points necessary to maintain data and decision integrity
- Implementation of all required integrations as informed by the previously mentioned points

  A Playback 2 deliverable does not constitute a fully functional solution that is ready for user acceptance testing. Additional elements that finalize the process implementation from Playback 0, the UI items from Playback 1, and the integration items from Playback 2 remain to be implemented.
- A focused demonstration of the previously mentioned deliverables, run from the Process Portal, implemented by the BPM and Technical Consultants, and delivered by the process owner with coaching from the BPM Consultant

The next steps after conducting Playback 2 use the understanding of the process from Playback 0, the data model and user interaction understanding from Playback 1, and the finished integration points from Playback 2, to focus on consolidating all themes into an end-to-end solution that is ready for user acceptance testing.

### Playback 3

*Focus* on consolidation of the previous themes and producing an end-to-end solution.

This playback is typically done at least once, and in concept it can be done as many times as is necessary to accomplish the theme of the end-to-end solution.

The *goals* of a typical Playback 3 are:

- Completing all necessary implementation details to consolidate the process automation, user interfaces, and integrations necessary to deliver a full BPM solution
- Delivering a fully deployable and testable solution that is ready for user acceptance testing

The BPM Developer defines and implements all remaining functionality points necessary to complete the end-to-end process solution. This final playback theme should not introduce any entirely new functionality to the solution. The focus should be on completeness, refinement, and stability.

The major *deliverables* of a typical Playback 3 include:

- An user testable solution, ready to be deployed to the user acceptance testing environment
- Documentation (beyond what is already built into the solution) needed to empower users, administrators, and system-level developers
- A clear description of all functionality that is deferred to the next revision of this project (after the current iteration is deployed to production)
- Implementation of all required functionality necessary for an end-to-end solution
- A focused demonstration of the previously mentioned deliverables, run from the user interface, implemented by the BPM Consultant, and delivered by the process owner with coaching from the BPM Consultant

The next step after conducting Playback 3 is to submit the project to the user acceptance testing phase, and prepare for a production deployment.

## 5.3.2 Authoring and style guidelines

Authoring and style guidelines empower large teams to collaborate more easily, with minimal time spent on specific guideline training.

The best guidelines make the intent of your fellow developers apparent from a cursory glance at their artifacts.

The worst guidelines are more concerned with consistency for the sake of itself, rather than the wanted outcome of consistency, which is increased productivity.

This section focuses on outcomes rather than pure tactics.

### What is a BPD

A Business Process Definition:

- Is a diagram that illustrates a business process
- Includes participants, steps, activities, and subprocesses

Objectives of a Business Process Definition are:

- Universally understood by both business and technologists
- Clearly and easily communicated in 5 minutes or less
- At any level of granularity
- Executable in a BPM System

### What is not a BPD

A Business Process Definition is *not*:

- An entity state diagram
- A collection of use cases or use case relationship diagrams
- A system relationship diagram
- An architectural diagram
- A workflow model (application development) or a screen flow

## Examples of bad process models

The following figures illustrate some of the most common modeling pitfalls.

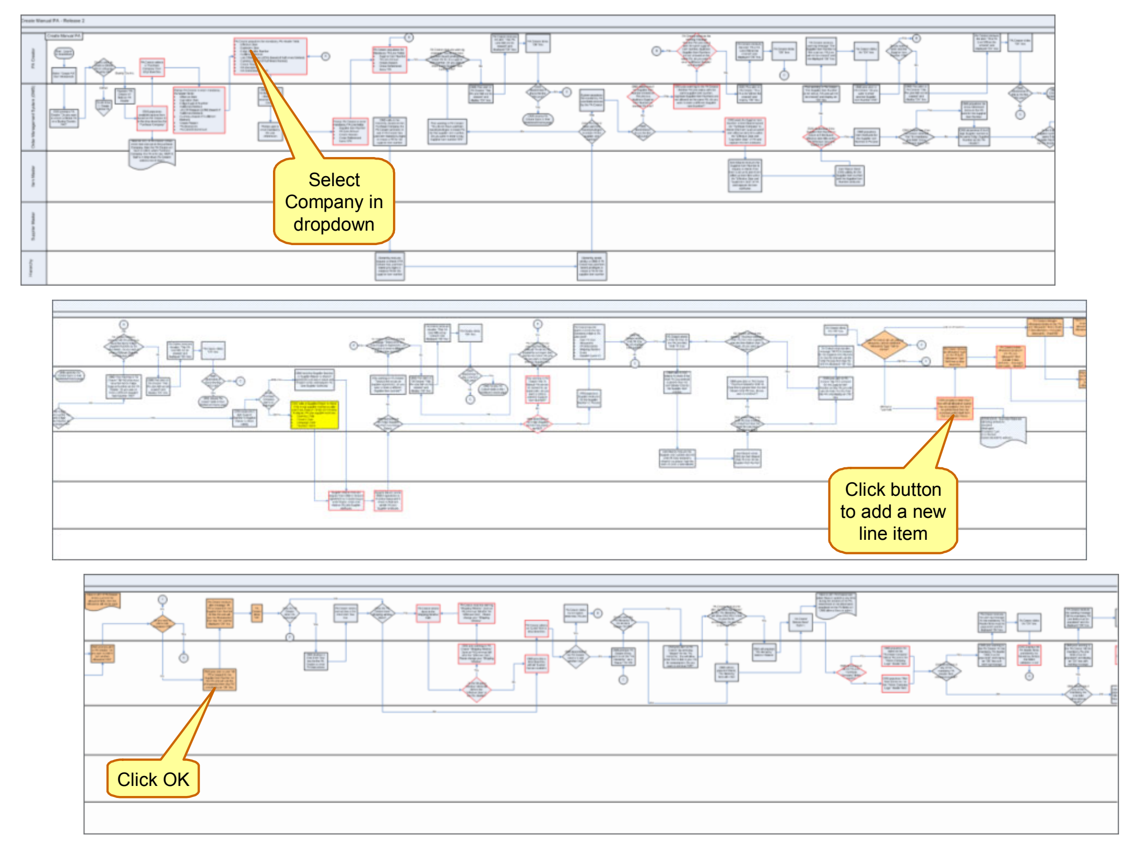

Select Company in dropdown
Click button to add a new line item
Click OK

*Figure 5-4 Process models that include the wrong type of detail*

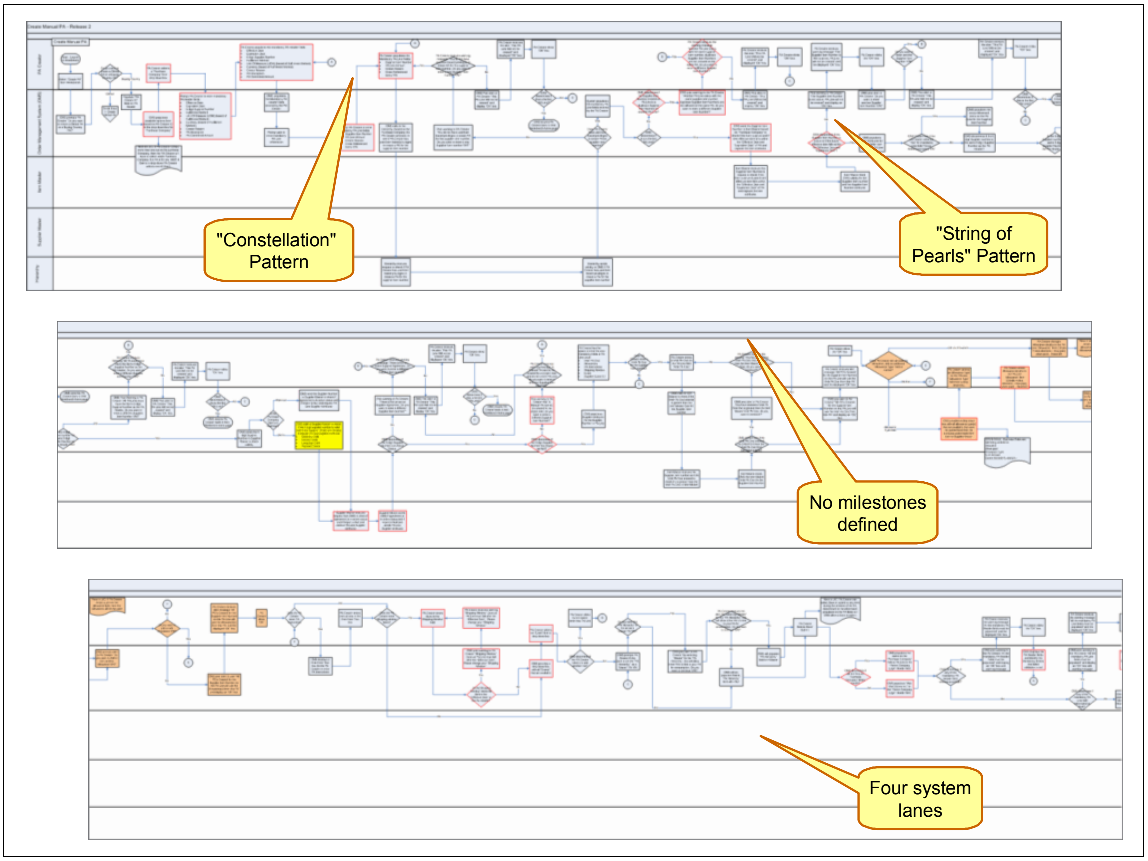


*Figure 5-5 Problematic modeling patterns*

Why these models are problematic:

- A process model includes tactical details such as clicking buttons or selecting options. Those details are service implementation details, as opposed to process steps, and should not be represented on a process diagram.
- A single process step should be a task done by a participant (likely, a human participant), after which the execution of the process is handed off to a different participant in a different swimlane.
- The *constellation pattern* illustrates a grouping of steps with a single flow line in and a single flow line out. These patterns are good candidates for abstraction into a subprocess diagram.
- Without milestones, we immediately “flatten” the view of the process.

- The large number of system lanes leads you to believe that there is "application logic" in this diagram (versus process flow/logic).
- The *string of pearls* pattern is a series of sequential tasks in the same swimlane, indicating a misunderstanding of the separation between the process layer and the service layer. These tasks should be condensed into single activities.

The fundamental difference between process models and services is that BPD activities are asynchronous. They are placed on the event manager for later execution or assignment. If you are expecting two units of work to occur immediately, you should use a service. In typical cases, reserve BPDs for task assignment and high-level process logic.

Table 5-1 provides high-level guidelines about when to use a business process definition (BPD) and when to use a service in IBM Business Process Manager.

*Table 5-1 High-level guidelines*

| Situation | BPD | Either/ both | Service |
|---|---|---|---|
| Display UI to a user. | | | X |
| Integrate to an external system (DB, rules engine, web service, and so on). | | | X |
| Background processing, scheduled processing. | | | X |
| Series of UIs directed at the same user. | | X | X |
| Simple Javascript. | | X | |
| Event correlation. | X | | |
| Exception handling. | | X | |
| Task handed from one user to another. | X | | |
| Split to parallel paths. | X | | |

## Five guidelines for better process modeling

Keep guidelines simple and easy to remember. Other guidelines might be relevant, but consider these five as the most important.

### *Rule of seven*

The “rule of seven” provides easy-to-remember guidelines:

- Keep the number of activities on your process diagram at any particular level of granularity down to seven or less.
- Beyond seven distinct activities on a process diagram, a business user loses the ability to understand the diagram or its intent.
- For anything beyond seven activities, strongly consider a subprocess to contain some of the detail.

### *Activity granularity*

At each level of abstraction, activities should be similar in scope and importance. For instance, avoid having an activity called “plan the party” that exists at the same level of abstraction as “open the door.” It might make technical sense, but it does not make business sense, and the process diagrams are meant to be readable by business users.

Leading indicators of mismatched activity granularity are the string of pearls and constellation patterns.

Remember that an activity in the process definition is a step in a process that can be implemented as a subprocess or a task. The definition or title of any task in a process definition should stop at task granularity. A task activity is a unit of work that a single participant (human or system) begins with the intent to complete.

An activity is a unit of granularity (step) in a process that has a goal that can be expressed as a singular outcome or output. It can be implemented as a task (human or system) or a subprocess. If it can contain multiple steps (screens in a workflow). These steps are not process steps. Lastly, a task can be a subprocess that is implemented as another BPD (one level lower) and follows the same guidelines as a parent BPD.

### *Activity description*

An activity should not be named after the context-less action being performed, like “look over application” or “add details.” Rather, it should be named to reflect the *goal* or output or *result* of the activity, such as “approve contract” or “adjudicate claim.”

Here is an easy formula to remember for naming your activities:

Activity name = action + entity

Or

[action verb] + [business object]

Avoid vague action verbs such as “process and perform [step].” Instead use action verbs that indicate a result or output. Use specific terms that business users recognize (even if they might be vague to others), and explain the terms in the activity description if necessary.

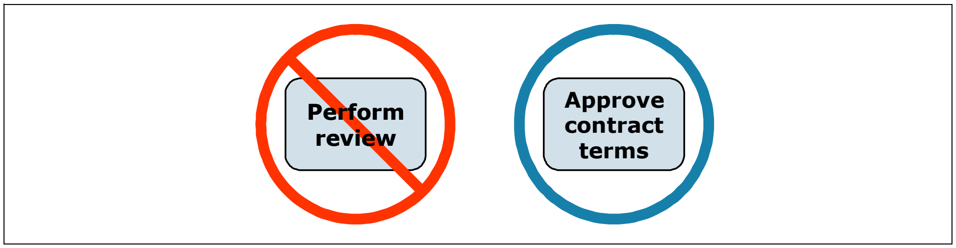


*Figure 5-6 Reflect the result or output in the name of an activity*

### Inputs and outputs

Inputs and outputs at the BlueWorks live level should be restricted to collections of business objects, such as candidate or claim, rather than individual fields that make up these collections.

Business entities should be defined from the business object model. When defining these entities, avoid specifying state for the entity (for example, signed contract) and specifying other qualifiers that are properties of the entity.

### The System lane

The system lane is meant to represent the BPMS and all of the external systems with which the BPMS interact. It can include activities performed by external systems but orchestrated by the BPMS.

The system lane should never include human activities or any activity that contains human interaction (screens). Always strive to avoid the string of pearls pattern in the system lane.

If there is much activity between the system lane and a participant lane, it could be because there is a low-level granularity (workflow and screen flow) being captured, where a few sentences in the description might suffice.

## 5.4 Design patterns

IBM Business Process Manager provides a number of generic artifacts for modeling purposes. These artifacts can be combined in various ways to solve various technical problems.

A design pattern is used to document how a combination of model artifacts can be used together to solve a specific technical problem. Design patterns are effective in promoting reuse and improving maintainability.

Unlike a framework or utility, a design pattern cannot be deployed directly. A design pattern is like having a pattern template for making clothes, where a tailor might use the same template to create a number of garments using different materials.

The design pattern provides a design template that empowers the BPM developer to solve the same technical problem consistently throughout the model, using different artifacts.

More information about a wide variety of design patterns in IBM Business Process Manager can be found on the IBM Business Process Manager Community Wiki at:

http://bpmwiki.blueworkslive.com/display/commwiki/Design+Patterns

## 5.5 Integration guidelines

IBM Business Process Manager V7.5 provides a number of ways to integrate with other systems. Although advanced integrations can always use the Integration Designer, you should first consider the ways in which Process Designer empowers you to implement integration with external systems. There are a number of standard ways of connecting with external systems within the Process Designer tool.

- Web services that use a supported WSDL can be consumed in minutes.
- Other connectors allow integration through REST/HTTP, raw SOAP envelopes, JMS message buses, and custom Java APIs.
- Process Designer includes built-in integration services that can be used for common external systems (SQL, SMTP, JMS, MQ, and so one)

You can find more information at:

http://bpmwiki.blueworkslive.com/display/commwiki/Integration+Technologies

## 5.6 Testing

Testing is an important part of any software project, and business processes are no different. However, testing a BPM solution is different from testing any other application development software.

The nature of BPM makes it a build-your-own-story type of system with many different paths and choices for the user. Additionally, it is largely a stop-and-go process with many different participants.

Automated testing in BPM primarily serves the purpose of unit and system testing. Any user interface or end-to-end business scenario testing must be done with real users, playbacks, and User Acceptance Testing (UAT.)

Implementation testing in IBM Business Process Manager can be handled in a number of ways, many of them that are similar to other software systems.

- Manual testing can be handled within BPM both as is, using the process inspector to run processes, and with some additional help from the process authors, by creating test services that allow a quality assurance team or the process authors to test individual coach screens and integration services.
- Automated unit tests in IBM Business Process Manager can be created as services that call another service with particular inputs and check the outputs, raising an exception if the output is unexpected. Automated functional tests can also be created using third-party tools.

It is highly suggested that unit tests be created for integrations, especially an Enterprise Service Bus (ESB) or database, because interfaces might change without author awareness. It is also a suggested practice to create manual test helper services for users coach views and search-type integration services. You can find more information at:

http://bpmwiki.blueworkslive.com/display/commwiki/Testing

## 5.7 General best practices

In addition to the specific topics described in this chapter (that is, playbacks, design patterns, methodology, and testing), there are a wide range of topics that have general best practices associated with them. General best practices include areas of exception handling, persistence, systems of record, and deployment. These and other topics are covered in greater detail on the IBM Business Process Manager Community Wiki:

http://bpmwiki.blueworkslive.com/display/commwiki/Best+Practices+Recommendations

# 6

# Deploying a process

This chapter introduces deployment concepts, procedures, and strategies for performing deployments in IBM Business Process Manager.

A complementary cookbook is provided on the IBM Business Process Manager Community Wiki site. The cookbook is living documentation that is continuously updated to reflect changes with new product releases and incorporate contributions from the user community. Site membership is at no cost. You can find the IBM Business Process Manager cookbook for deploying business processes on the IBM Business Process Manager Community Wiki at:

http://bpmwiki.blueworkslive.com/display/commwiki/Cookbook+for+deploying+a+business+process

The cookbook includes information about:

- The deployment of process applications to both online and offline servers
- The management of the runtime servers (online and offline) from the Process Center
- The deployment of service applications and modules
- The migration of running process instances during deployment

## 6.1 Overview of core concepts

This section introduces the core concepts around deployment in IBM Business Process Manager and helps you to get answers to the following questions:

- *Where* do you deploy?
- *When* do you deploy?
- *What* do you deploy?
- *How* do you deploy:
  - A release?
  - A test fix?

This section also describes the importance of a release and installation strategy. This strategy is an important aspect of deployment, and it is of even more important when you move from a single Business Process Management (BPM) or a few BPM projects to a comprehensive BPM program.

## 6.2 Process center

After modeling, designing, and implementing a business process solution, the next step in the business process application lifecycle is the deployment of the application to a runtime server.

The Process Center is the central repository in IBM Business Process Manager 7.5. The Process Center repository contains the artifacts of process applications and allows the sharing of processes and assets across process applications. It also provides governance and lifecycle management capabilities and plays an important role in the deployment of process applications (Figure 6-1).

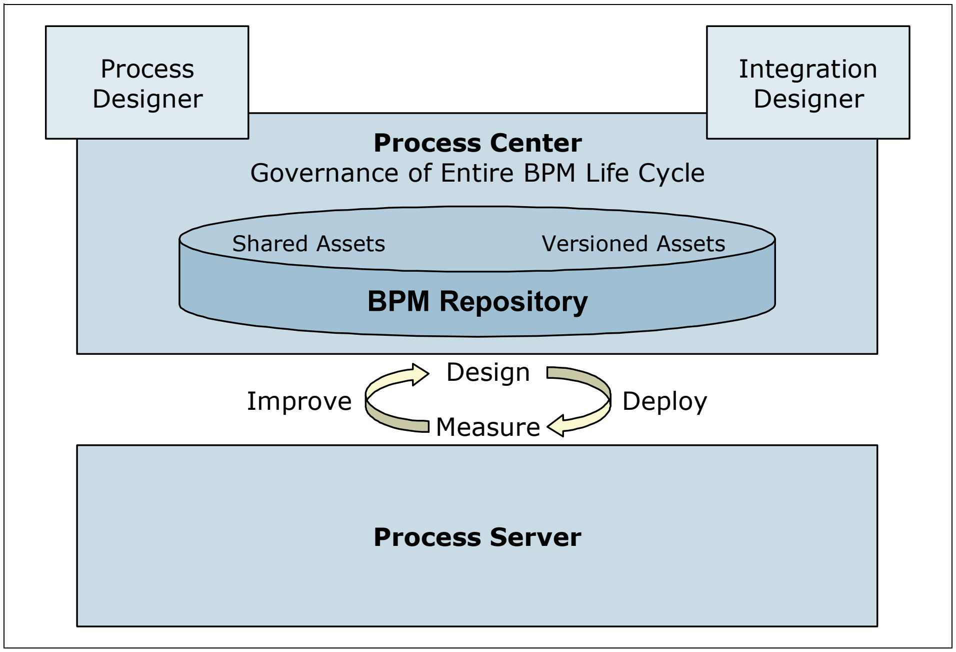


*Figure 6-1 Governing process applications with the Process Center*

The centralized deployment provided in IBM Business Process Manager 7.5 eases the tracking of process application versions deployed to any runtime process server.

Governance and lifecycle management are important aspects of BPM and are of special importance when you think about scaling BPM across your enterprise, going from a BPM project to a BPM program.

In the context of deployment, the Process Center is used for the following tasks:

- Management of runtime process servers

  The Process Center maintains a list of online and offline process servers that represent your deployment target. These servers can be classified as test, staging, or production servers.

- Deployment to online process servers

  The Process Center console is used to deploy snapshots of process applications to process servers (test, stage, or production servers).

- Creation of deployment packages for offline-server deployments

  From the Process Center, you can create deployment packages for process application snapshots. Deployment packages are required if you want to deploy to an offline process server.

## 6.3 Where do you deploy

This section provides an overview of the various environments in IBM Business Process Manager V7.5 and answers the question: *Where* do you deploy?

IBM Business Process Manager supports the concept of environments. Each environment is used for a specific purpose that is determined by the type of the environment. Figure 6-2 shows a typical setup of IBM Business Process Manager with a development, test, staging, and production environment.

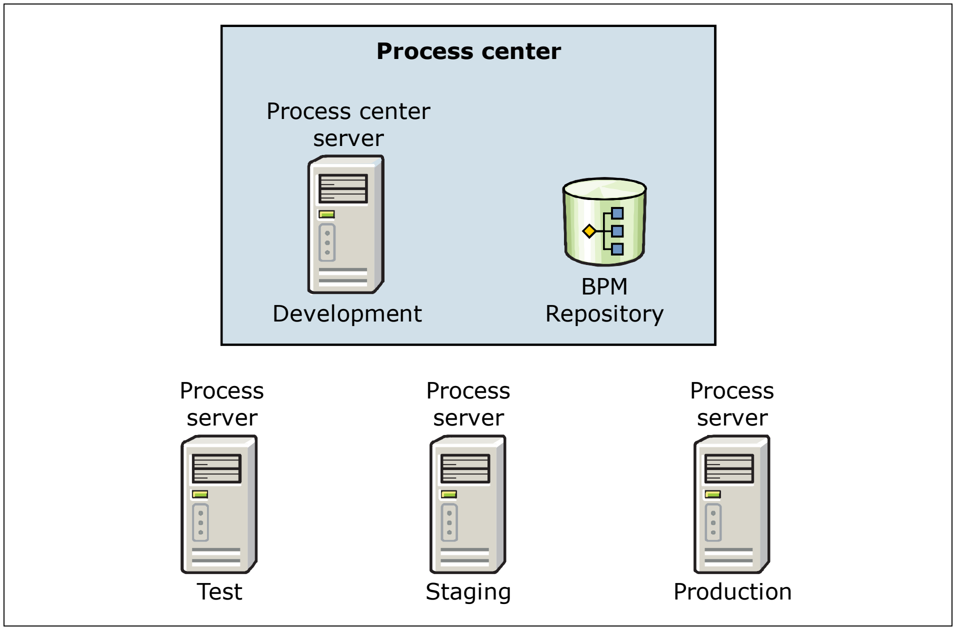


*Figure 6-2 Environments*

The Process Center is a core component in IBM Business Process Manager, as it contains the central BPM repository. It is the only environment where artifacts of Process Applications and Toolkits can be created and edited.

## 6.3.1 Process Center server (development environment)

As shown in Figure 6-2, the Process Center contains a Process Center Server. This server acts as the *development environment* and is automatically assigned the development environment type.

The development environment is the only environment where you cannot specify the environment type, and in contrast to other environment types, you can have only one environment as the development environment.

The Process Center Server is intended to be used during application development by the BPM solution developers. It is important that you use this environment according to its purpose and that you *do not* use it for hosting your production process applications.

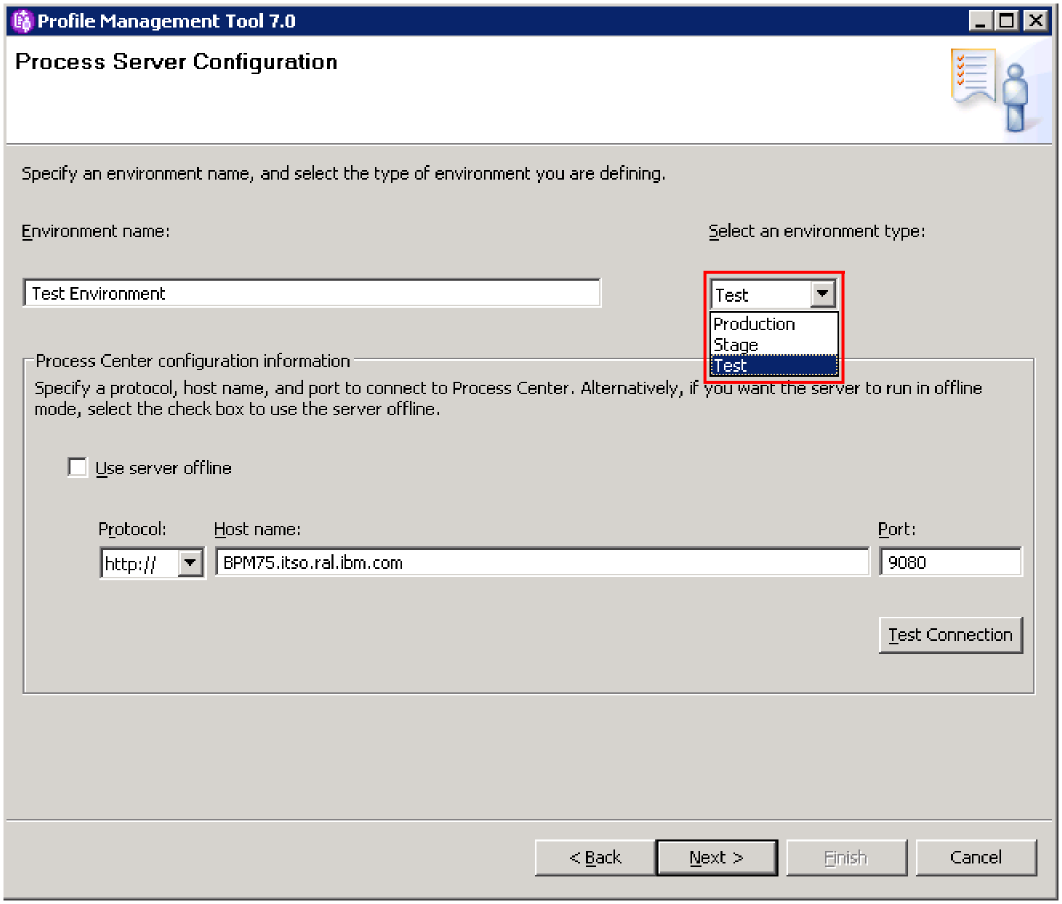
### 6.3.2 Process servers (test, staging, and production environment)

Apart from a development environment, a typical BPM setup contains one or more runtime servers. It should at least contain a production environment, but usually contains additional environments for special purposes.

Figure 6-2 shows three different runtime servers (*process server*), each of which serves a different purpose. For runtime process servers, the following environment types can be designated:

- Test
- Stage
- Production

Figure 6-3 shows the assignment of the environment type to an environment during the profile creation in the Profile Management Tool.



Profile Management Tool 7.0

Process Server Configuration

Specify an environment name, and select the type of environment you are defining.

Environment name: Test Environment

Select an environment type: Test

Process Center configuration information

Specify a protocol, host name, and port to connect to Process Center. Alternatively, if you want the server to run in offline mode, select the check box to use the server offline.

□ Use server offline

Protocol: http://

Host name: BPM75.itso.ral.ibm.com

Port: 9080

*Figure 6-3 Environment type selection during profile creation*

> **The manageprofiles utility:** If you create your profiles using the `manageprofiles` command-line utility, then you specify the environment type using the `-environmentType` parameter. For a detailed list of the supported parameters, go to the following address:
>
> http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/topic/com.ibm.wbpm.ref.doc/topics/rins_manageprofiles_parms.html

Unlike the Process Center Server, Process Servers need to be associated with the Process Center during or after the creation of the server (profile creation). Runtime servers can be configured as:

- Online server/connected server

  Online servers are directly connected to the Process Center. Application deployments to online servers can be performed from the Process Center console. Usually, test and staging environments are configured as online servers.

- Offline server

  Offline servers are not connected to the Process Center. Usually, production environments are configured as offline servers.

## 6.4 When do you deploy

The third question in the context of application deployments is: *When* do you deploy? Depending on the type of environment, deployments either occur automatically or need to be performed manually.

### 6.4.1 Automatic deployments during development

When modeling and implementing a business process, BPM analysts and solution developers use the Process Designer and Integration Designer tools to create the solution. These tools are connected to the Process Center that contains the development environment: the Process Center Server.

Deployments to a Process Center Server occur automatically when:

- You run, or “play back,” a process in Process Designer (Figure 6-4).

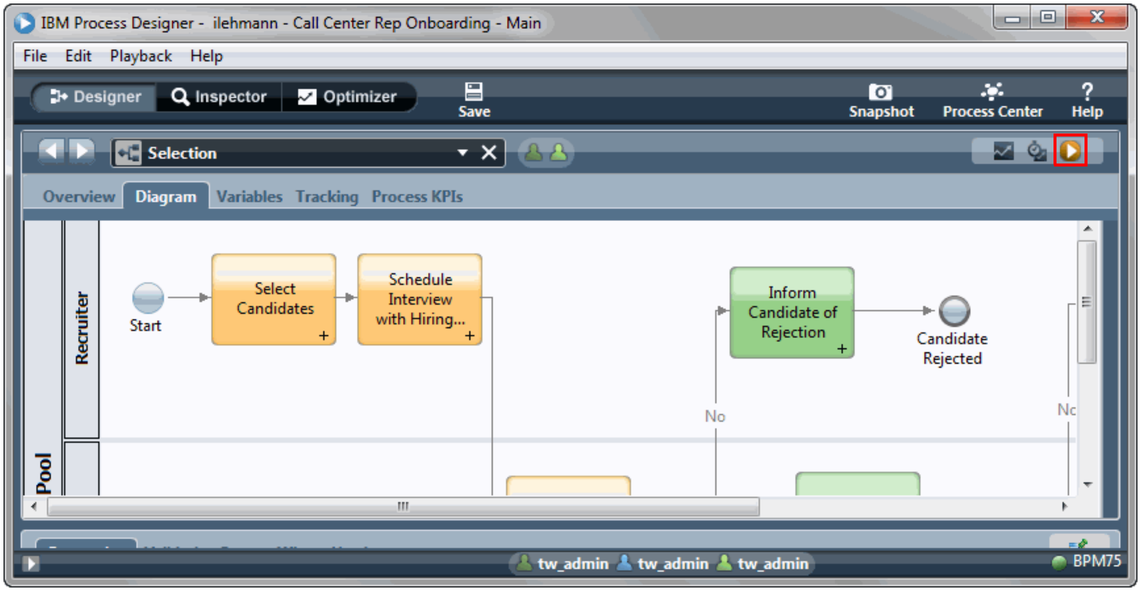


*Figure 6-4 Run a playback from IBM Process Designer*

- You change dependencies in Process Designer, such as add a dependency, remove a dependency, or change the version of a dependency (Figure 6-5).

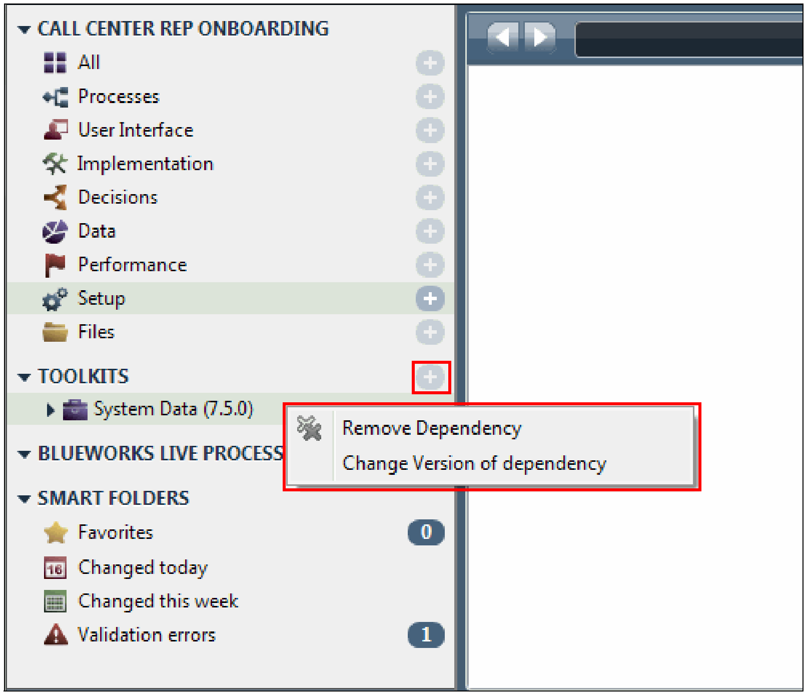


*Figure 6-5 Change dependencies*

- When you publish the application from Integration Designer (Figure 6-6).

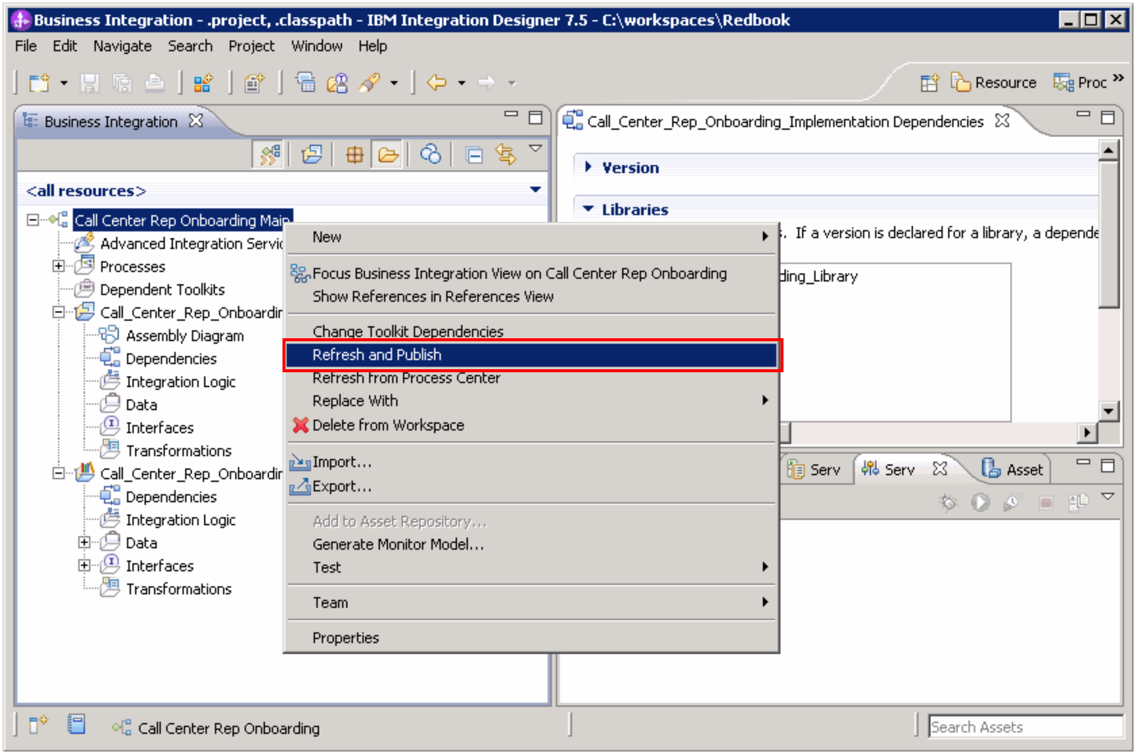

Business Integration - .project, .classpath - IBM Integration Designer 7.5 - C:\workspaces\Redbook
File Edit Navigate Search Project Window Help
Resource
Proc
Business Integration
<all resources>
Call Center Rep Onboarding Main
Advanced Integration Servic
Processes
Dependent Toolkits
Call_Center_Rep_Onboardir
Assembly Diagram
Dependencies
Integration Logic
Data
Interfaces
Transformations
Call_Center_Rep_Onboardir
Dependencies
Integration Logic
Data
Interfaces
Transformations
Call_Center_Rep_Onboarding_Implementation Dependencies
Version
Libraries
. If a version is declared for a library, a depende
ding_Library
New
Focus Business Integration View on Call Center Rep Onboarding
Show References in References View
Change Toolkit Dependencies
Refresh and Publish
Refresh from Process Center
Replace With
Delete from Workspace
Import...
Export...
Add to Asset Repository...
Generate Monitor Model...
Test
Team
Properties
Serv
Serv
Asset
Call Center Rep Onboarding
Search Assets

*Figure 6-6 Publish application changes in IBM Integration Designer*

## 6.4.2 Manual deployments to runtime servers

In contrast to the development environment, where deployments to the Process Center Server are performed automatically, deployments to runtime Process Servers need to be initiated manually.

Specific versions (snapshots) of a process application are deployed to a runtime server at a specific point in time. This process is done either from the Process Center console for deployments to online Process Servers or through scripts for deployments to offline Process Servers.

## 6.5 What do you deploy

The third question in the context of application deployments is: *What* do you deploy? This section introduces the important terms and concepts in this context.

The deployment cookbook on the IBM Business Process Manager Community Wiki describes the deployment steps in detail.

### 6.5.1 Process application

In IBM Business Process Manager V7.5, applications are developed as process applications by the BPM solution developers. A *process application* is a container for process models and their implementations. It can contain the following artifacts:

- Process artifacts (from Process Designer)
  - Business process models (business process definitions)
  - Coaches
  - Services that are required to implement activities or to integrate with other systems
  - Toolkits that the process application depends on
- Integration artifacts (from Integration Designer)
  - SCA modules
  - SCA libraries
- Monitoring artifacts (from Integration Designer)
  - Monitor models
- Other items required to run the process

### 6.5.2 Process application snapshot

Process application snapshots record the state of the items within a process application at a specific point in time and represent a specific version of your process application. The snapshot contains all the components that are part of the process application and the dependent toolkits.

Process application snapshots can be created from the Process Center console or in the Process Designer view. The snapshots are managed from the Process Center console and represent the deployment unit that gets deployed to online and offline process servers. You can deploy process application snapshots to process center servers (development environment) and to process servers (test, staging, or production environment). To deploy to a stand-alone process server, you must take a snapshot of the process application, because this artifact is the only artifact that you can deploy to process servers.

**Snapshots:** Besides snapshots for process applications, you can also create snapshots for toolkits. Toolkit snapshots are *only* deployed when you deploy a process application snapshot that depends on this toolkit snapshot. This way, the owners of the process applications can decide whether they want to use the new version of the toolkit.

## 6.5.3 Business level application

If you have IBM Business Process Manager V7.5 Advanced and your process application contains not only artifacts from Process Designer but also artifacts from WebSphere Integration Developer, then a *business level application* (BLA) is created under the covers. It is a container for the process application and its assets (for example, monitor models). In addition, each snapshot of such an advanced process application has its own BLA.

**Further information about BLAs:** The concept of BLAs is a concept of WebSphere Application Server. For more information about BLAs, go to the following address:

http://publib.boulder.ibm.com/infocenter/wasinfo/v8r0/topic/com.ibm.websphere.nd.multiplatform.doc/info/ae/ae/crun_app_bla.html

You do not see BLAs in the Process Center, but many of the administration tasks for a snapshot, such as stop or start, are technically done at the level of the BLA. This action allows for a quicker and simpler administration of the snapshot and all of its assets.

BLAs are also relevant in the context of deployment. BLAs that represent the tip of a process application or that represent a concrete snapshot of a process application can be deployed.

**Managing BLAs:** BLAs are created automatically for process applications/process application snapshots under the covers. Do *not* manage BLAs manually using the WebSphere Application Server administrative console.

## 6.5.4 Tip

The tip is a special snapshot containing the most recent content of the process application. When your BPM analysts and solution developers change a process application, the changes are automatically saved to the Process Center repository at the tip of the working track. This situation means that the tip is the only snapshot where you can change contents. This configuration is useful during development, and for that reason the tip can be run only on the development environment (the Process Center Server). Tips cannot be deployed to all other environments (to Process Servers).

The deployment of a tip to a Process Center Server happens automatically in your development environment when people change the artifacts. For this reason, this chapter focuses on the deployment of process application snapshots.

## 6.5.5 Track

Tracks can be compared to branches in version control systems. Tracks allows parallel development to occur with isolation from changes in other tracks. Tracks are especially important in the context of fixing problems in a version of a process application that is already deployed to production. By using tracks, the development on a new version can continue in parallel to fix the problems in an older version. Tracks do not support merging capabilities. When a developer fixes a problem in a track, the fix also needs to be applied to the main track so that it is included.

## 6.5.6 Putting it all together using an example

Figure 6-7 shows the Process Center repository with different artifacts. It contains two process applications with different snapshots. Each process application snapshot represents a deployment unit and can be deployed to a Process Server.

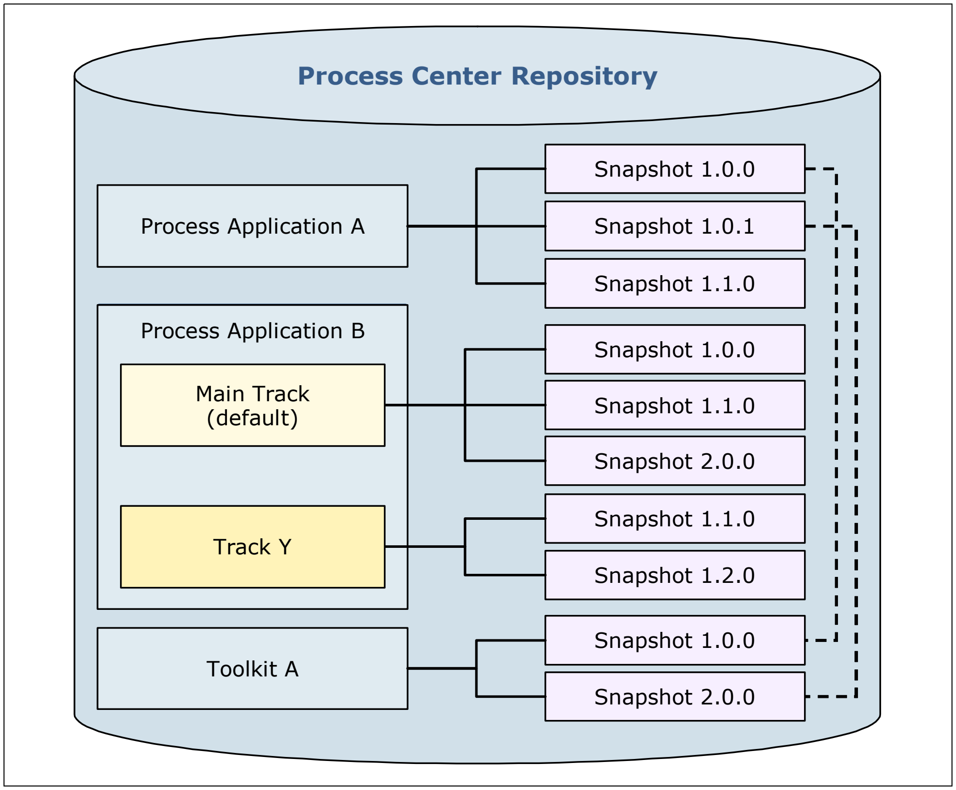


*Figure 6-7 Process applications, toolkits, and snapshots*

While process application A exists only in the main or default track, an additional track (track Y) was created for process application B for bug-fixing purposes. Snapshots for both tracks of process application B exist and can be deployed to a server.

The process center repository also contains a toolkit (toolkit A) that has two versions or snapshots. The dotted lines in Figure 6-7 show the dependencies between the snapshots of process application A and toolkit A:

- Snapshot 1.0.0 of process application A uses snapshot 1.0.0 of toolkit A.
- Snapshot 1.0.1 of process application A uses snapshot 2.0.0 of toolkit B.

# 7

# Managing a process

This chapter focuses on the management aspects in Business Process Management (BPM.) Management requires visibility so that opportunities for change can be surfaced. The exercise of designing processes inherently requires you to be prepared for change.

This chapter begins with an overview of how to measure and analyze business applications and processes through metrics, key performance indicators (KPIs), and service level agreements (SLAs). The next step is to use the outcome to further refine process models for better performance as part of a continuous improvement of the business processes. Then the chapter describes how to move from a BPM project to a BPM program. The last section of this chapter describes the tools for managing business processes and process applications.

## 7.1 Overview of measuring business processes

Many companies start with BPM from a workflow perspective with automation in mind. Automating tasks that are labor-intensive or prone to risk because they are manual is an important part of BPM, but it is only part of the BPM value proposition. The measuring capabilities provided by IBM Business Process Manager help you move beyond the traditional ideas of process automation, system analysis, and process re-engineering. We want to advance our focus to the next level of BPM maturity: process optimization.

Optimization is about achieving dramatic results though business process management. The goal of optimization is to enable continuous process improvement (CPI) through key BPM principles. Improvement means change. As humans, we naturally resist change, and in an organization, resistance to change can be even more difficult to overcome. In the spirit of improvement and optimization, we need to educate ourselves and each other, and overcome the fear of change. We must instill the idea of CPI with positive messages, such as embracing change and allowing change. Change is good.

We are not just aiming to improve one project within an organization. We aim to implement a BPM solution by giving the organization visibility into its business processes. Successful organizations do not only care about automation and orchestration opportunities. They also care about process improvement in terms of metrics, SLAs, KPIs, and other ways to quantifiably measure their improvement. Often, this perspective is a new one within an organization, and it can be difficult to get the business owners to identify the metrics that they care about and to articulate how best to measure the success of their business processes. This conversation is a critical one to conduct at the beginning of your project, because it empowers you to prove the benefit of the BPM solutions effectively.

The monitoring of business processes should be part of the development lifecycle from the beginning. During the discovery phase, the business analyst specifies the metrics necessary to achieve the goals of the company. The measurement process starts with the business analyst to define the KPIs and SLAs, including definitions of what to measure and which information is shown to whom.

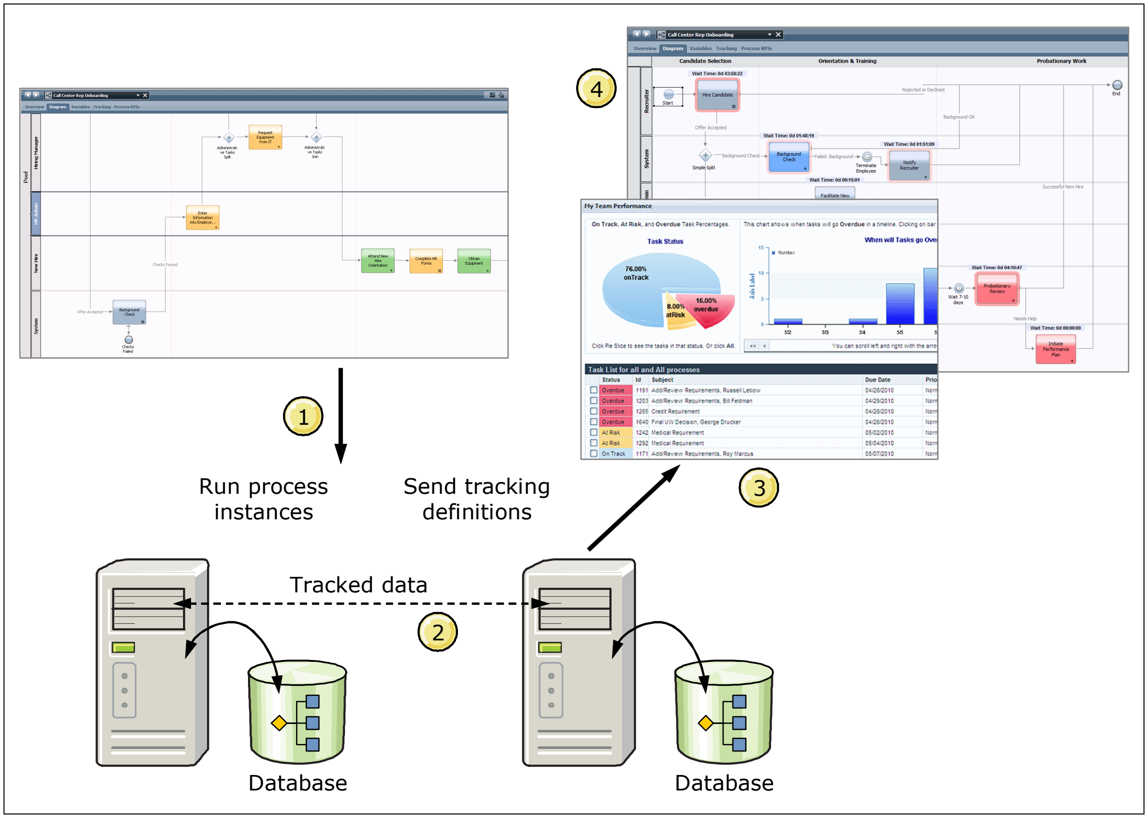
This first section outlines the core concepts in measuring business processes in IBM Business Process Manager:

- Metrics specified on the business process to gather information
- Reporting and optimization capabilities

In this narrative of control and visibility, the Performance Data Warehouse is the cornerstone.

### 7.1.1 Performance Data Warehouse

IBM Business Process Manager provides built-in and dynamic process visibility for BPMN flows with the Performance Data Warehouse. Figure 7-1 illustrates the Performance Data Warehouse architecture.


4
1
Run process instances
Send tracking definitions
3
Tracked data
2
Database
Database
My Team Performance
On Track, At Risk, and Overdue Task Percentages.
Task Status
76.00% onTrack
8.00% atRisk
16.00% overdue
Click Pie Slice to see the tasks in that status. Or click All.
When will Tasks go Ove
5/2
5/3
5/4
5/5
You can scroll left and right with the arro

Task List for all and All processes

| | Status | Id | Subject | Due Date | Prio |
|---|---|---|---|---|---|
| □ | Overdue | 1191 | Add/Review Requirements, Russell Lebow | 04/28/2010 | Norn |
| □ | Overdue | 1203 | Add/Review Requirements, Bill Feldman | 04/29/2010 | Norn |
| □ | Overdue | 1265 | Credit Requirement | 04/28/2010 | Norn |
| □ | Overdue | 1640 | Final UW Decision, George Drucker | 04/28/2010 | Norn |
| □ | At Risk | 1242 | Medical Requirement | 05/02/2010 | Norn |
| □ | At Risk | 1292 | Medical Requirement | 05/04/2010 | Norn |
| □ | On Track | 1171 | Add/Review Requirements, Roy Marcus | 05/07/2010 | Norn |

*Figure 7-1 The Performance Data Warehouse*

The flow is:

1. A single BPMN model drives both the business activity monitoring and the execution of business processes.
2. Performance Data Warehouse automatically gathers and correlates the specified process metrics and business data during the execution of all human and system activities. This data is analyzed continuously.
3. Real-time visibility into tasks and process performance is provided through dashboards and reports.
4. Process Optimizer provides a crystal clear view into in-flight or historical performance bottlenecks, visualized as *hot spots* on the process diagram.

The Performance Data Warehouse is essentially a data warehouse dedicated to the performance data collected during the run time of your processes. Performance Data Warehouse gathers data asynchronously from the process server or process center databases. This process can be configured.

Because all performance-related metrics are stored in Performance Data Warehouse, you can use third-party business intelligence tools to query the database and generate reports.

## 7.1.2 Metrics

To determine which metrics to capture in the Process Data Warehouse, we follow a top-down approach (Figure 7-2).

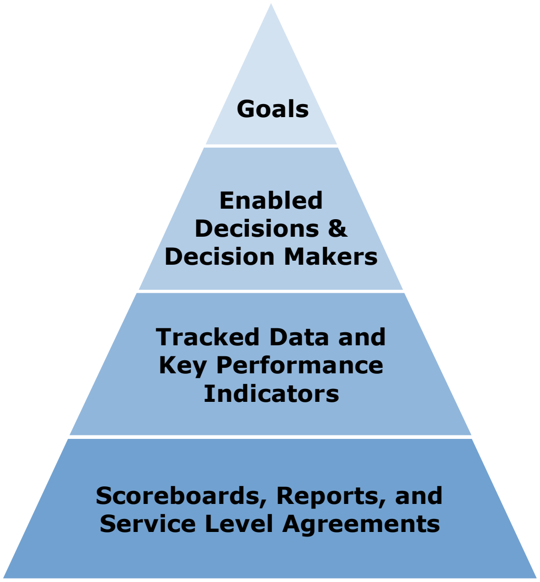

Goals
Enabled Decisions & Decision Makers
Tracked Data and Key Performance Indicators
Scoreboards, Reports, and Service Level Agreements

*Figure 7-2 Goals pyramid*

In this top-down approach, first analyze your process goals, and then determine which metrics to capture to gauge whether you are meeting process goals. If the metrics do not help you determine whether you are meeting process goals, you probably should not be concerned with tracking those particular metrics.

Following this approach enforces a discipline of making sure that your metrics actually align with process goals and empower your decision makers to determine whether the process is meeting its goals.

**Tracking process efficiency and effectiveness:** “The metrics used to track process efficiency and effectiveness may differ significantly from the data used to maintain the state of the process.” – Derek Miers[a]

[a] Source: http://queue.acm.org/detail.cfm?id=1122688

Another way to look at this process is from the perspective of whether the metrics and KPIs, as defined, are actually helping decision makers guide the process toward meeting program goals. There are decision makers at every level in an organization, from the board of directors to night shift supervisors. These people make decisions that stimulate change throughout the organization. All of these decisions affect organizational change in one or more categories:

- Resource changes
  - Human resources.
  - Non-human resources.
- Process changes
  - Add or remove an activity.
  - Change a decision gateway.
  - Raise or lower a threshold.
- Business changes
  - Add or remove products or services.
  - Add or remove markets.

Recognizing that *managing a process* can more astutely be characterized as changing one or more of these three categories often makes it easier to identify those metrics and KPIs that decision makers need to make sound business decisions. There is no reason to create reports that do not serve a purpose. By identifying the decisions that need to be made, you can ensure that the information that you track and report on serves a purpose. Find the decision makers, document their decisions, and then identify the metrics and KPIs that they need, and the reports practically write themselves.

All too often, accurate and meaningful metrics are not tracked because even though the correct questions were asked, the decision makers were not in the proverbial room.

After the metrics are defined, they can be implemented in the Process Designer tool. There are two approaches to tracing the business data for Performance Data Warehouse:

- Autotracking
- Manual tracking

Based on the tracked data gathered through these two approaches, KPIs and SLAs can be defined for the business process. In the following sections, we describe these concepts in more detail.

## Autotracking

Enabling autotracking occurs at the level of a business process definition (BPD). When you enable autotracking for a BPD, you automatically track KPIs associated with your BPD and the activities in the BPD. These KPIs include both default and custom KPIs. KPIs act as conditions to trigger SLAs.

The second use of autotracking is to track business data throughout your BPD. Autotracking includes tracking points at the entry and exit of every step in your BPD (for example, activities and services).

When first implementing a process, it is good practice to enable autotracking because it allows the use of KPIs and SLAs. Having this data from the beginning provides valuable insight into your process as it evolves. Also, the data captured by autotracking is used by the Optimizer for continuous process improvement. Therefore, autotracking is enabled by default on all BPDs in IBM Business Process Manager V7.5.

With autotracking enabled, data is being traced at every step in the process. Autotracking impacts performance and storage cost, especially where large amounts of business data are included. An alternative to tracing business data is using tracking groups and timing intervals, which allow you to be more precise and record only the data when and where needed.

Special consideration should also be given to BPDs that are designed to run infinitely. Autotracking such BPDs can produce large result sets that might cause memory issues on the Performance Data Warehouse.

In general, it is important from the beginning to consider how long you want to keep tracked information available in the Performance Data Warehouse, and to establish an archiving procedure.

For our fictitious Call Center Company C, keeping the duration of the onboarding process as short as possible is key. Enabling autotracking provides management with valuable information about the time that each step in the process takes, and this information can be used for process improvement. Therefore, it makes good sense to keep autotracking enabled for the main process *call center rep onboarding* and all its subprocesses *selection*, *background check*, and *complete HR forms*.

## Manual tracking

Autotracking is a “sweeping” approach to capture business data in your processes. The alternative approach of *manual tracking* gives you more control over which business data to track, and at which points in your processes to create more advanced custom reports. Using the manual tracking method, you combine *tracking groups*, *tracking points,* and *timing intervals*.

With tracking groups, you define which business data is related to each other, and then analyze that data using custom reports. The specific locations in your BPDs where data needs to be captured are indicated with *tracking points*. A tracking point allows you to associate the specific business data with the variables defined in the tracking group. The third concept, *timing intervals*, helps you analyze the amount of time elapsed between specific steps in your process.

Besides allowing you more control over capturing data, tracking groups provide you with the means to track data across multiple processes and process applications. For processes within the same process app, create a tracking group and define the tracking points in as many BPDs as you want. If you want to capture data from multiple business processes across different process apps, create a tracking group in a toolkit, and then refer to that toolkit from each process app where you would like to use the tracking group.

### Key performance indicator

A key performance indicator (KPI) is a broad concept whose exact meaning should be agreed upon with the client as you begin the BPM project. Within IBM Business Process Manager, the KPI concept is a business metric related to the business process or an activity within the business process (such as wait time or resource cost). To track KPIs for your processes, autotracking must be enabled. Tracking data for KPIs allows you to analyze process performance in the Optimizer (see 7.1.5, “Enabling decisions with the Optimizer tool” on page 211). It also supports SLAs. IBM Business Process Manager provides standard KPIs and a facility to create custom KPIs.

The standard KPIs provided by IBM Business Process Manager V7.5 are:

- At process level
  - Total time (clock)
- At activity level
  - Cost
  - Execution time (clock)
  - Labor cost
  - Resource cost
  - Rework
  - Total time (clock)
  - Value add
  - Wait time (clock)

A KPI can also be rolled up into a higher-level aggregate KPI. When creating a KPI, you can associate it with the aggregate KPI and specify a weight factor. This situation is useful when creating SLAs. For example, the *resource cost KPI* and *labor cost KPI* roll up into the *cost KPI.*

In our example, Call Center Company C wants to measure the average cost of the onboarding process for candidates who are hired. This KPI comprises two separate concepts to track:

- The total cost for each activity in the process
- Whether the candidate is hired

Custom KPIs can also be defined. Interesting data that would qualify as good custom KPIs are business data that you want to include in an SLA.

Call Center Company C is interested in knowing the number of people who are rejected based on a failed background check. As this activities is costly and time-consuming, management wants to be informed of the rate of failure. If the rate is low enough, the process can proceed with the administrative tasks and the orientation activities without waiting for the result of the background check, thus saving valuable time. But a sudden increase in the failure rate might indicate a problem with either the selection process or the background check activity itself.

## Service level agreement

As with KPIs, the concept of an SLA is broad and goes beyond the functionality provided by the SLA component within IBM Business Process Manager. When talking about SLAs for your project, several components should be considered, each with specific benefits:

- Due dates
- Timer event
- SLA component

### *Due dates*

If your requirements for an SLA concern time and you do not want or need to have an automated response to a violation, due dates are your first option. Due dates can and should be specified on activities and for the entire process. The due date is used by the default scoreboards to show the tasks and processes on target, at risk, or overdue (see 7.1.3, “Empowering decisions through reporting” on page 209, for more details). The task owners and their managers have clear visibility into the status, empowering them to act when needed.

### *Timer event*

As the name indicates, timer events are time-based events. A timer event differs from a due date in that it allows for immediate and automated action upon the violation of the condition, and a time event is related to a single activity. The SLA artifact also provides for automated action, but the action is not executed at the exact moment at which the condition is violated.

### *SLA component*

The SLA component is an aggregate measure of KPI performance across instances. The SLA component provides for both temporal (time related) and non-temporal conditions. SLAs can be based only on default and custom KPIs, and not on timing intervals. SLAs allow you to trigger a consequence based on the violation of a condition of one KPI for one or more activities. This consequence can be an email notification or the start of a new business process. For example, you can send an email notification if the average duration of the onboarding process exceeds the expected 30 days.

When using SLAs, remember that the consequences are triggered when the associated activity starts or completes and *not* when the condition is violated. Besides allowing action, SLAs are used for reporting (see 7.1.3, "Empowering decisions through reporting" on page 209) and performance analysis (see 7.4, "Managing processes" on page 222).

SLAs can be defined on any KPI measure for one or more activities, or on the entire process.

As mentioned earlier, Call Center Company C is interested in knowing the number of people who are rejected based on a failed background check. The business owner wants to monitor the level of background failures, and the owner wants to act if the number rises significantly because of its high impact on the cost of the process. This requirement was translated into a specific SLA:

Only up to *5%* of all the new hires can be rejected due to a background check.

SLAs are defined on a KPI associated with an activity. It is possible to define in a single SLA a condition that can be applied on several activities, as in the example, but you cannot define a KPI to measure one metric across multiple activities. For example, if you want to define an SLA on the total time of the three sequential activities (*facilitate new hire's orientation*, *attend call center training*, and *activate new hire)*, group these activities in to a subprocess, and then measure the total time needed for that subprocess.

It is important to have new employees integrated into the business and being productive as soon as possible. The call center training is planned to start on day 4 of a new employee's arrival. The training is a significant cost, and the employee needs equipment to reap optimal benefits from the training. Therefore, the necessary equipment must be issued by the end of day 3.

This brings us to our second SLA, *employee setup time*:

The *facilitate new hire's orientation* subprocess must be finished within three days from the start date of a new employee.

In concrete terms, this situation means that the *total time KPI* for *facilitate new hire's orientation* activity cannot exceed three working days.

### 7.1.3 Empowering decisions through reporting

As soon as you determine the key metrics to support the process goals and empower decision makers to improve the process, you need to consider how to best present the tracked information to the decision makers. IBM Business Process Manager provides scoreboards to help you quickly craft the report presentation that you want.

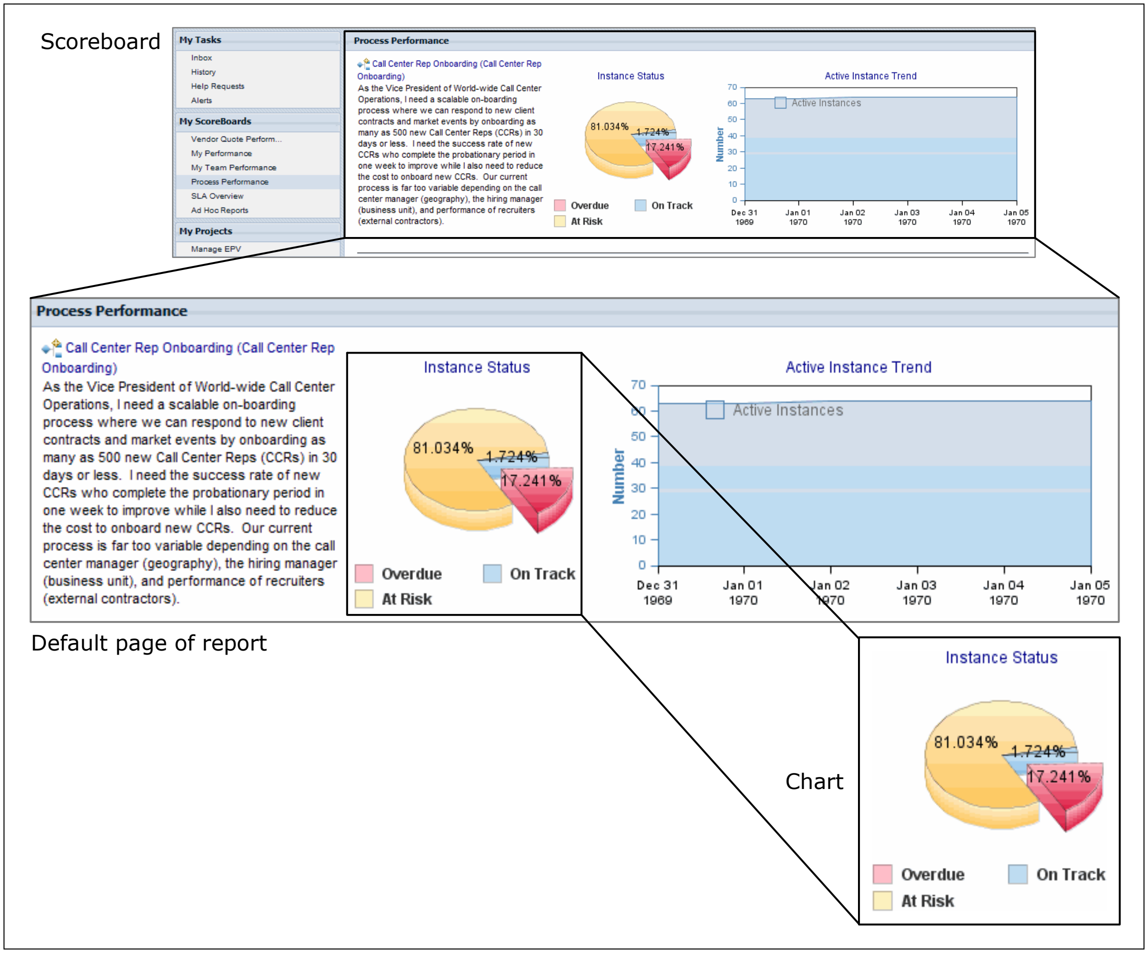
Scoreboards should:

- Give visibility into the performance of the process.
- Allow decision makers to validate that the process supports the business goals.
- Help predict future performance.
- Empower decision makers.

The most important thing is to create scoreboards early and to demonstrate them frequently in cross-functional playbacks. From the results of the process discovery for the onboarding process, it is clear that the reporting should contain information about the average cost to hire a new employee and the total time of the process. The report could also show the success rate of the recruiters, such as which recruiter has the highest percentage of hired employees versus proposed candidates. This information allows Call Center Company C to optimize its process by preferring to work with the most successful recruiters.

## Core concepts in reporting

The key component in reporting is the *scoreboard*. Scoreboards are made up of *reports*. Each report consists of one or more *pages*, with one or more *charts* on each page. The charts get the information from *data sources*. These concepts are illustrated in the default Process Performance scoreboard in Figure 7-3.


Scoreboard

My Tasks
- Inbox
- History
- Help Requests
- Alerts

My ScoreBoards
- Vendor Quote Perform...
- My Performance
- My Team Performance
- Process Performance
- SLA Overview
- Ad Hoc Reports

My Projects
- Manage EPV

Process Performance

Call Center Rep Onboarding (Call Center Rep Onboarding)

As the Vice President of World-wide Call Center Operations, I need a scalable on-boarding process where we can respond to new client contracts and market events by onboarding as many as 500 new Call Center Reps (CCRs) in 30 days or less. I need the success rate of new CCRs who complete the probationary period in one week to improve while I also need to reduce the cost to onboard new CCRs. Our current process is far too variable depending on the call center manager (geography), the hiring manager (business unit), and performance of recruiters (external contractors).

Process Performance

Call Center Rep Onboarding (Call Center Rep Onboarding)

As the Vice President of World-wide Call Center Operations, I need a scalable on-boarding process where we can respond to new client contracts and market events by onboarding as many as 500 new Call Center Reps (CCRs) in 30 days or less. I need the success rate of new CCRs who complete the probationary period in one week to improve while I also need to reduce the cost to onboard new CCRs. Our current process is far too variable depending on the call center manager (geography), the hiring manager (business unit), and performance of recruiters (external contractors).

Default page of report

Chart

*Figure 7-3 Reporting overview*

### *Scoreboard*

The scoreboard is the way through which reports are shown to the user. Scoreboards appear in the user's process portal if the scoreboard is exposed to the user. Each scoreboard can show one or more reports. An example of a default scoreboard is My Performance.

#### *Report*

A report can have multiple pages, but there must be a single default report page. The default page typically displays multiple charts together in a single page. The chart drilldowns control the navigation to the other report pages.

#### *Chart*

Charts define the way in which the data retrieved through the data source is shown to the user.

#### *Data source*

A data source consists of integration services, data transformation, and filters. The integration services determine the data selected for the chart. The data transformation runs the integration services and transforms the result to fit the requirements of the chart. Filters are used to limit the amount of data in a chart. The filter is displayed in the report and allows the business user to modify the filter value and refresh the report.

### 7.1.4 Enabling decisions through flexible processes

We recognized in 7.1.2, “Metrics” on page 203 that managing a process can best be described as empowering managers to make decisions, and decisions mean change. In the previous section, we described how to support these decisions with meaningful reports. A second consideration is to anticipate these changes and make the process flexible to allow for changes without the need to redeploy the model. The Exposed Process Variables (EPVs) are a powerful concept designed to empower business users to change variables used in the process *while* process instances are running. The advantage of using EPVs is that they are global in nature, meaning that changing the value of an EPV impacts all processes or service that uses it.

Not every change can be anticipated and controlled through EPVs. In the end, adhering to BPM best practices as opposed to standard application development ensures that the process itself is flexible and easily adaptable to a changing business environment.

### 7.1.5 Enabling decisions with the Optimizer tool

The Optimizer tool is the IBM Business Process Manager tool for continuous process improvement (CPI). The Optimizer provides various analysis scenarios, ranging from simple simulations to validate your overall process modeling strategy, to advanced *what-if* comparative analyses.

When you run instances of a BPD with autotracking enabled, IBM Business Process Manager tracks and stores data for configured KPIs in the Performance Data Warehouse. IBM Business Process Manager uses stored KPI data when you run certain types of historical analyses in the Optimizer, but not all historical analyses available in the Optimizer rely on data generated and stored due to KPIs.

The Optimizer tool simulates your processes while you are developing them to understand how well those processes might perform. The Optimizer tool runs simulations using estimates that you provide for staffing levels, activity execution times, and so on. Simulating your processes during development allows you to test and refine process designs before implementation. In the Optimizer tool, you can analyze your processes after they are running for a while, using historical data stored in the Performance Data Warehouse (for example, to identify bottlenecks and other issues through color-coded heat maps. See Figure 7-4.

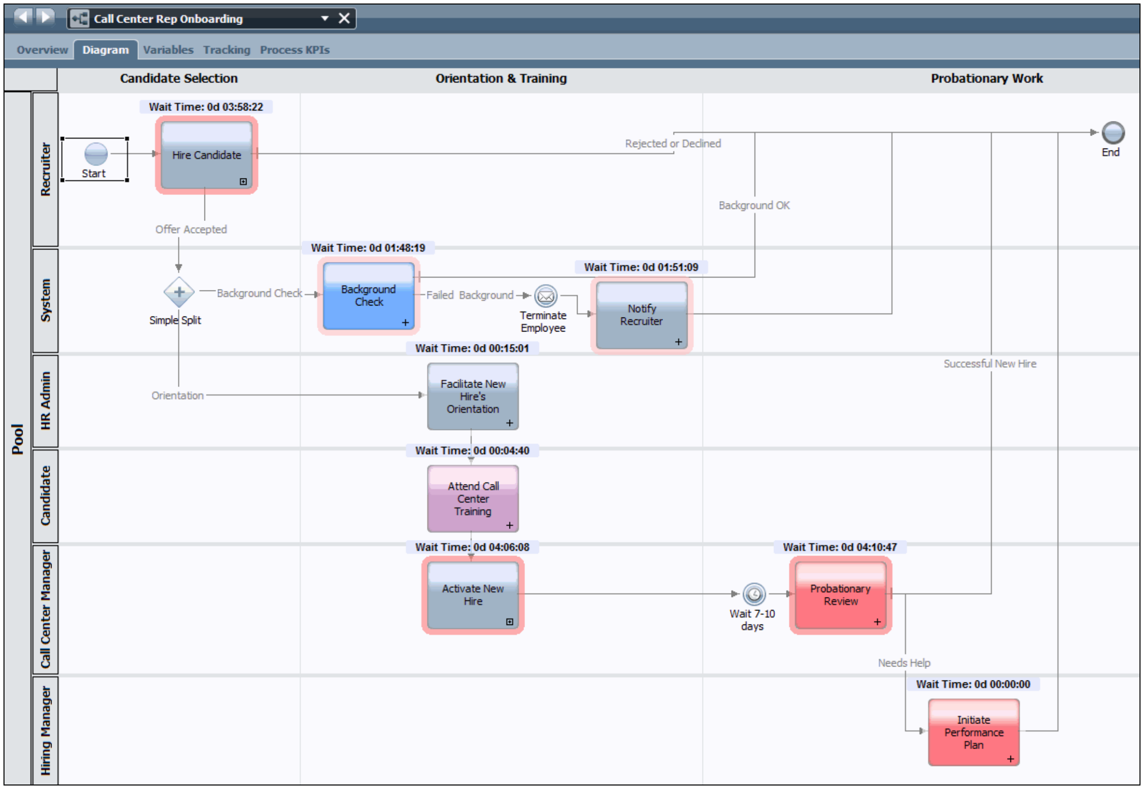


*Figure 7-4 Optimizer tool heatmap*

The Optimizer tool also provides recommendations for problematic activities to address issues found in the process. For each process that you want to analyze, you need to enable autotracking (see “Autotracking” on page 205). Running historical analyses allows you to measure and then improve the efficiency of your processes.

In the onboarding process for Call Center Company C, a simulation was run to validate the decision to already start the orientation and training of the new hire before the background check is found. To validate this decision, an SLA is created to check the percentage of failed background check. With the Optimizer tool, the assumptions taken for the simulation can be checked against the historical data.

## 7.2 Improving business processes

This section introduces a general approach to improving business processes.

### 7.2.1 Walking before you run

Before we delve into how to examine process data for change, let us step back and talk about what generally happens after process owners see that they can get crystal-clear visibility into their processes. Picture yourself in the typical BPM project where it takes about 90 days to go into production. Generally, if you take the approach described in 7.1.2, “Metrics” on page 203, you have excited stakeholders thinking of all kinds of possibilities for metrics and reports. Make sure that you keep the reins tight. Aim to keep the scope limited to only those metrics and reports that indicate to the process owner whether their process is succeeding or failing.

Why? The fact is, your view is limited as to how the process actually performs after it is in production. You really do not yet know how long each activity is going to take to complete. You do not yet know how long the process takes to complete on average. You do not yet know how well your users respond to a process approach. There are many unknowns. You approach the project believing that the process that you design helps meet program goals, but you really do not *know* whether this situation is actually the case until your process is running in production for one or two months. The Optimizer tool (see 7.1.5, “Enabling decisions with the Optimizer tool” on page 211) can help a great deal in determining how your process runs. But real data always trump simulations.

Make sure that you stay within the defined scope, and do not let stakeholder requests run too wild. Let the process run in production for a while to capture your key metrics, and then determine when and what to change.

### 7.2.2 Indicators of change

In 7.1.2, “Metrics” on page 203, we described how to determine what metrics to capture by using a top-down approach to analyzing your process goals, and then determining the metrics that serve to validate that those goals are being met. (As stated earlier, if a metric does not help you determine whether you are meeting process goals, you probably should not be capturing it.)

After you ensure that your metrics align with process goals, you can start to analyze your process for areas of improvement. Now we can begin to describe typical improvement indicators.

Most processes have typical improvement indicators that you can focus on to determine when and what to change in your process. Typical improvement indicators include:

- Process not contributing to program goals.
- Inefficiencies in the process.
- Costs are too high.
- Too much rework.
- Cycle time is too high.

The last four indicators are what are talked about when people begin to focus on process improvement indicators. All of these indicators require that you know what you want to measure. Even with cycle time, which is a fairly simple measure requiring no configuration, the process owner or someone needs to know what the goal is to have a baseline to measure against. We are going to focus on the first typical indicator of the “Process not contributing to program goals.” The other four are fairly obvious.

The metrics that you define in your initial and ongoing analysis work should help you make decisions that align with program and corporate goals. If you have done your groundwork correctly, your metrics align with program and corporate goals. It is critical that these goals are aligned if you expect the metrics that you glean from your process to point you toward a “true north” in improving your process.

Figure 7-5 shows a simple depiction of an analysis hierarchy that you can use in determining whether a process issue is impacting your process or program goals, enabling you to prioritize.

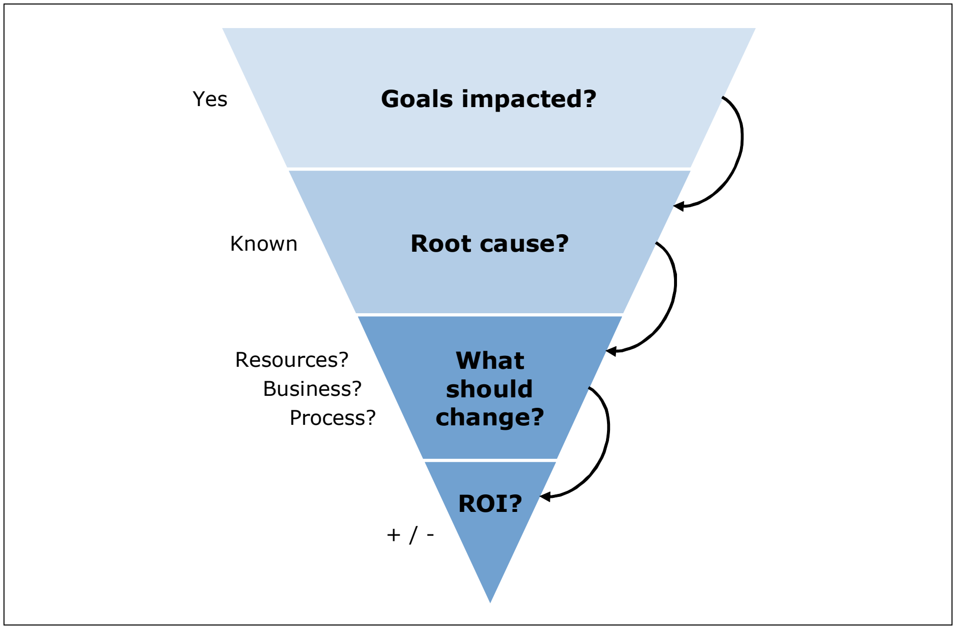


*Figure 7-5 Analysis hierarchy*

While this situation is a simple representation of what goes on when analyzing your process for improvement, it plays an enforcing function. The analysis hierarchy makes you focus on metrics and process issues from a process and program goals perspective.

## 7.2.3 Does the issue impact process goals

With the analysis hierarchy, the first thing that you should ask yourself as the process owner or decision maker is: *Does this issue impact my project or process goals?* If the answer is no, you can put it at the bottom of your list of process improvement priorities. If it is not impacting your program and process goals, why focus on it? Your time is limited and valuable. If the answer is yes, you should progress to root cause analysis.

### 7.2.4 What is the root cause

The root cause might be obvious from the process metrics and other information that IBM Business Process Manager gives you. More often than not, though, you need to do some research to discover what the root cause is. In any problem solving methodology, it is important that the root cause of the issue be firmly established before determining what, if anything, should change.

Let us take the example in the onboarding process of the decision to offer a new candidate a position. The process might be reporting a significant percentage of refused candidates. This situation could be happening for many reasons. Perhaps the same recruiters propose candidates who do not match the job profile. Perhaps the hiring managers are too stringent in their evaluation. Or, perhaps the minimum salary that the candidates expect is higher than Call Center Company C foresaw in its decision criteria. As you can see, there can be many possible causes for a process inefficiency. Root cause analysis is a crucial step before progressing to the work of designing a solution.

### 7.2.5 What should change

After you determine the root cause of the issue, you need to determine what should change to resolve the process issue. Generally, there are three things that can change to affect a business:

- Resources
- Business
- Process

As a process owner or stakeholder, when you discover that your process is not helping you meet program goals, and you know the root cause, you need to ask yourself what should change. Is it people, process, or the business that needs to change? If it is your business that needs to change, this change is a major effort and beyond the scope of this book, and probably beyond the scope of you as the process owner or stakeholder. If it is a resource issue, as a process owner or stakeholder, you might have the means to make change happen. The great thing about IBM Business Process Manager is the ability to run the numbers through the Optimizer and actually determine how manipulating your process resources would impact your process. The other powerful benefit of the Optimizer is that you can take this solid data to your management team to support your request for a change in resources. This benefit cannot be overstated. It is one thing to go to senior management and *say* that adding two resources to this process would help a great deal. It is a different matter, and far more palatable to senior management, to take the hard numbers that the Optimizer calculates and tell management exactly what would happen if the company increases the resources for this process.

The third question is, What should change in the process? Any of the common indicators for change that we described earlier could be reasons for changing the process. Your root cause analysis helps determine what is wrong, and guide you to what needs to change in the business process. It could be as simple as the business rule change that we described in the previous paragraph, or as complex as redesigning a subprocess.

### 7.2.6 Continuous improvement

When you go through a few iterations of this process, the question of "When do I stop improving my process?" inevitably comes up. In a perfect world, with perfectly unlimited resources for your company, the answer is never. You keep improving your process until it no longer needs improvement, which probably never happens. The reasons that it never happens is that the competition is always improving, or at least attempting to, and the business conditions of the world around your business are always changing. As an example, try being a purely brick-and-mortar book store these days. You find that the world changes and that the model does not work anymore.

Stepping back to the perfect world and unlimited resources concept, there are a few companies that, during the last decade, acted as though they had unlimited resources, and those companies are the ones that you hear about in the bankruptcy news. The fact is, it is not a perfect world, and your resources are never unlimited. So, you must use the analysis hierarchy, or a variant there of, to determine when and how much you should change your processes. The question really comes down to what is the most efficient allocation of your limited resources right now? As a process owner and a key contributor to corporate strategy, you must analyze when it makes sense to keep changing your process and when it makes sense to leave it alone because your limited resources are better spent elsewhere.

In summary, start small, and make sure that you keep the scope reigned in during the initial development of your process metrics. Make sure that any metrics and reports that you build for the process actually help you determine whether you are meeting your process goals. After you are in production, use the analysis hierarchy to determine whether your process needs to change, what needs to change, and whether it is worth fixing the issue. When you reach a high confidence level with doing this analysis with one process, begin to apply the methodology to other processes within the same program or value chain.

## 7.3 From project to program

While monitoring your process is important on its own, processes do not exist on their own. They are part of a larger set of interconnected processes that make up the core business of a company. If you consider monitoring at a project or process level, you are missing out on the real power of Business Process Management and process visibility. Your goal should be going from a single process up the value chain to the enterprise level.

The logical starting point when beginning with BPM within your organization is the project of single process level because you want to reduce risk and build competency in this new field. The strength of BPM comes into play when you start to plan from the top down. The following sections show how to elevate the discussion within your organization and hopefully enable your company to become truly process driven. The approach described is just one of many possible approaches, and we are not going to reinvent the wheel. However, experience from many BPM engagements shows that this approach works, so we share it with you so that you might benefit from it.

### 7.3.1 Rolling up KPIs from project to program

The next step in our approach is rolling up process KPIs to a higher level. This process is one way to achieve the goal of elevating all the way from project to process to enterprise level. To begin with, you look for common metrics between different processes. Your best candidates are metrics such as cost and time because they are almost always measured in any process. This situation assumes that our first business process is running in production for a few months. You are now starting to work on your next business processes and trying to extend the visibility across all processes, using common performance indicators.

From a business perspective, our fictitious Call Center Company C needs to know how much time it takes from signing up a new customer to staffing all the required positions and how this time can be reduced. At a higher level, this information would be provided by the average time to fill an open position, that is, from opening a new position to having a new hire actively engaged.

The average total time measured in the onboarding process rolls up into the average total time with a roll-up multiplier depending on the success rate of your onboarding process. The success rate is the number of people interviewed compared to the number of people successfully hired.

Figure 7-6 illustrates the three levels of metrics. The way to determine what metrics to capture does not change. The goals pyramid does not change. You must start by evaluating what is important to measure across multiple processes, and not just the one process or project.

- The process metrics (M3) level represents the process metrics that roll up to metrics that are shared by multiple processes. For example, take the total time of the onboarding process of Call Center Company C.
- The subprocess metrics (M4) level represents the metrics that you need to capture in each of your subprocesses to roll up to your process metrics. This metric can be represented by the time needed to execute the subprocess selection.
- The same goes for the activity metrics (M3) level. You need to capture the key metrics at the activity level to be able to roll these metrics up to your subprocess and process metrics.

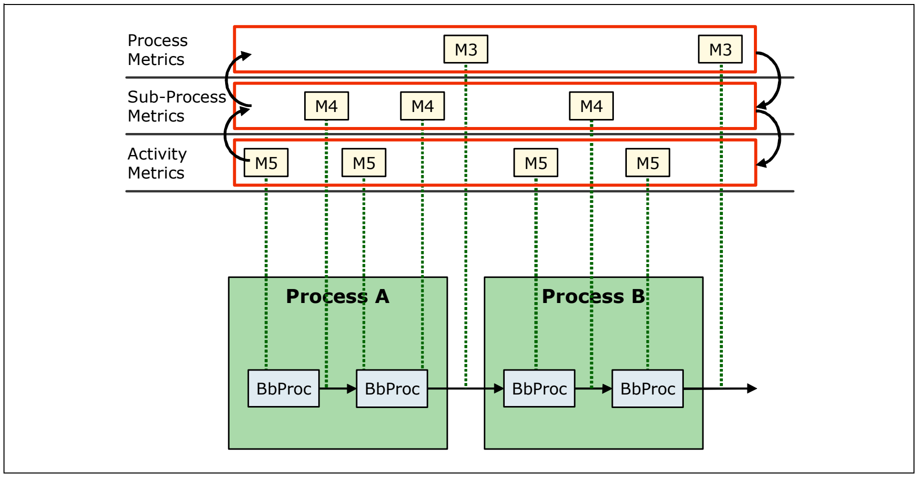


*Figure 7-6 Rolling up metrics*

Viewing this situation from a top-down perspective, you start from the goals and determine what metrics need to be captured from each process in the program to measure whether you meet the goals. Then at the subprocess and activity (task) level, you need to determine what metrics need to be captured to roll up to the process level. After you have up this set, your managers are empowered to see what processes are contributing to or hindering your goals.

You can then bring it to the next level of rolling metrics up to the corporate level (Figure 7-7).

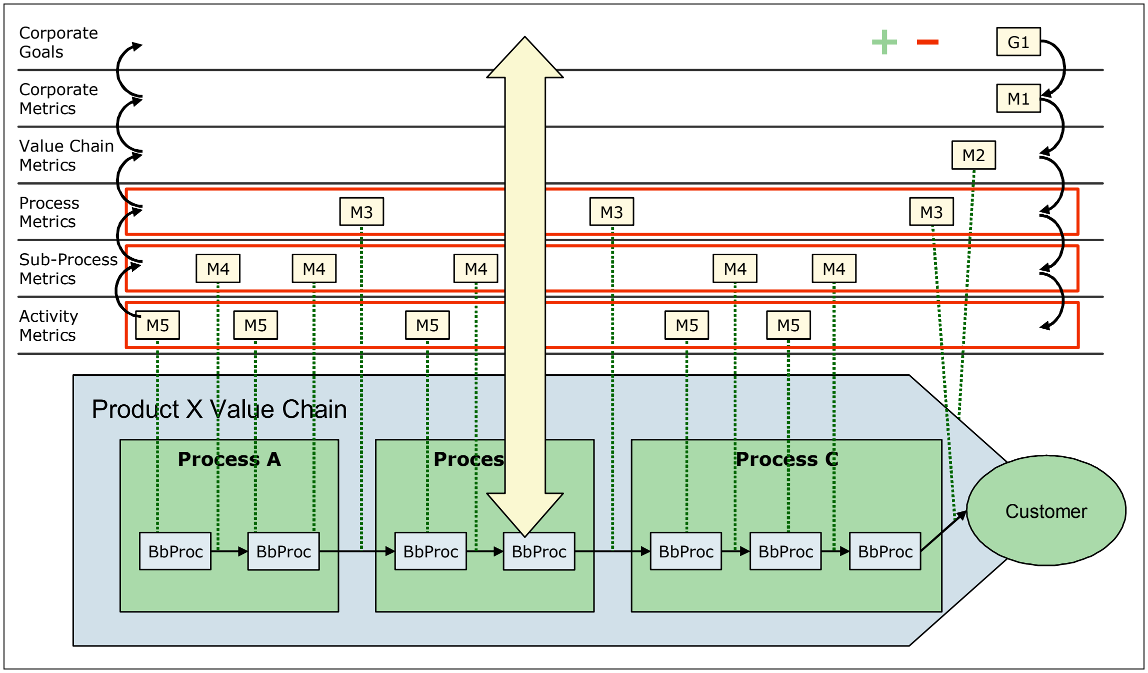


*Figure 7-7 Rolling metrics up to corporate goals*

The basic premise of this approach is that you start at the top level with your corporate goals (G1). Every company exists to fulfill its corporate goals. The most basic and obvious goal is producing shareholder value, but there are other goals, such as gaining a certain percentage of market share, increasing customer satisfaction, decreasing cost, increasing employee productivity, and so on. Many of the goals change on a yearly basis, depending on performance of the previous year and other market factors. After you set these corporate goals, all the metrics rolling up to those goals should be tailored to indicate success or failure in meeting those goals (represented by the plus and minus signs in the Corporate Goals lane).

As multiple processes within a particular value chain run, you roll all of the metrics up, and because you took a top-down approach, each of them should inform decision makers about where and how processes are helping or hindering corporate goals.

After it is implemented, this approach enables you to adjust at the activity level (M5), which improves some performance metric for an activity, and by doing so, impacts corporate goals (G1). The result is a direct linkage between the tasks that people are doing and the goals of the organization. Key management now has visibility into the processes and clearly understands the impact on the corporate goals within a value chain. Process owners can then make small, incremental changes and see how they are impacting process goals *and* corporate goals. IBM Business Process Manager gives you the tools to quickly change your processes, and a view of key performance indicators to determine success of failure.

The following sections describe how to approach changing your process.

## 7.3.2 End-to-end monitoring

The Business Activity Monitoring (BAM) provided by IBM Business Process Manager provides built-in and automatic visibility of processes defined in the BPMN model in Process Designer. When your goal is to extend BAM to end-to-end visibility across business processes, you have:

- Processes that span organizational boundaries with independent project life cycles.
- Processes implemented with a combination of a BPM system and existing applications or infrastructure.
- Processes that are not yet implemented in a BPM system. End-to-end visibility drives priority of improvements.

You can achieve this goal easily through enabling business monitoring support for IBM Business Monitor. With business monitoring, you can report on events from BPMN and BPEL processes and SCA and mediation events, which can be specified in the IBM Integration Designer tool. You can also combine events from external systems for end-to-end business operations.

Activating business monitoring with IBM Business Monitor can be done at two levels within IBM Business Process Manager V7.5:

- Generated monitor model: Monitors BPMN events only (modeled in Process Designer)
- Custom-developed monitor model: Monitors BPMN and other BPM events (modeled in Integration Designer)

### Process Designer

Enabling process monitoring through IBM Business Monitor is performed on a process application level. This action generates a monitor model. The monitor model comes with an auto-generated dashboard for each process application in business space. The generated monitor model subscribes to the events emitted by all BPDs contained in the process application and any referenced toolkits. For more information about the tracked data, see 7.1.2, “Metrics” on page 203.

### Integration Designer

In the Integration Designer tool, you can create a custom monitor model to monitor your BPMN, BPEL, SCA, and mediation events, or started products. The starting point for this custom model is the monitor model generated from the process application. During the generation step, you can fine-tune the events to include in the generated models.

The monitor project can be associated with a process application in the Process Center and published online.

## 7.4 Managing processes

This section describes the options available for managing processes and process applications. Management of process apps is primarily done through the Process Center Console. Additionally, the Process Admin Console provides access to administration services defined in the process app (EPVs can be managed through the Process Admin Console), enables monitoring of active processes, and queues and supports post-deployment activities.

## Process Center Console

The Process Center contains a repository for all process applications, toolkits, process services, and other BPM assets. The Process Center Console is the management tool for the Process Center Repository (Figure 7-8). The Process Center supports the entire governance lifecycle within BPM. The Process Center provides a shared development environment and centralizes process deployment visibility and control across all environments (see Chapter 6, "Deploying a process" on page 183).

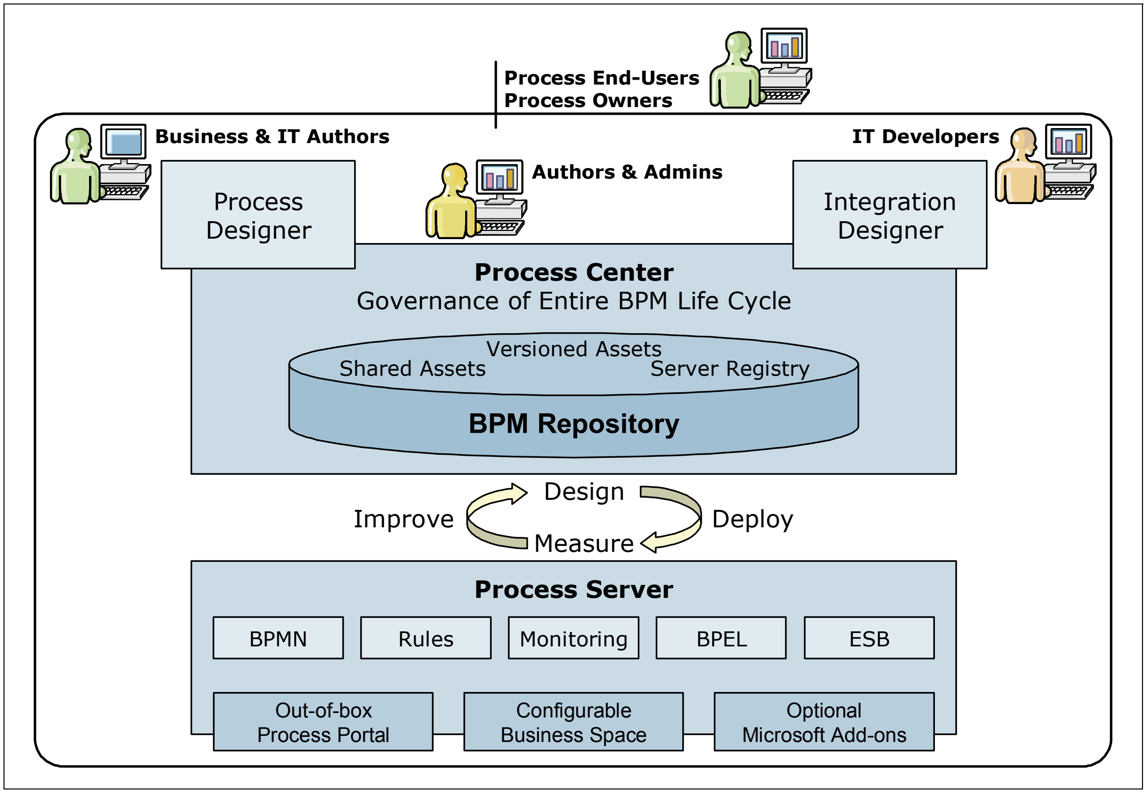


*Figure 7-8 Process Center repository*

The Process Center Console enables you to:

- Manage assets and dependencies (for example, process app, toolkits, snapshots).
- Manage servers in the various environments.
- View the deployment dashboards: See the in-flight instances of the deployed snapshots of the process app.

**More information:** For more information, go to the following address:

http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/index.jsp?topic=%2Fcom.ibm.wbpm.admin.doc%2Ftopics%2Fwelcome_wps_adm.html

# 8

# Business process governance

This chapter introduces business process governance.

## 8.1 Introduction

Establishing *business process governance* should be a foundational part of an organization’s business process management (BPM) planning beginning with the first BPM project. Putting in place the building blocks for a business process governance model helps an organization establish a vision for the overall BPM effort. Governance also avoids difficulties when the organization seeks to scale from the original BPM projects to a more transformational BPM program.

During the BPM journey, certain common challenges are likely to emerge. Having adequate business process governance in place (aided by the capabilities of IBM Process Center) helps in managing and addressing those challenges.

The typical challenges stem from one of three sources:

- Many authors

  How will you promote reuse and achieve synergies between several design and development teams working in parallel to deliver BPM projects?

- Many processes

  In a modern enterprise, there tend to be hundreds of process models, in various modeling tools with each modeler employing different modeling style, and each model possessing multiple versions. Proceeding in such an environment puts pressure on an organization’s capability to keep its automated process solution lifecycles short and responsive, and obscures the BPM stakeholder’s ability to provide a clear overview of where the projects are heading.

- Many assets

  As the number of process projects increases, the number of assets and interdependencies increases, leading to increased complexity, insufficient process reuse, increased complexity, and loss of disciplined governance.

These three sources are illustrated in Figure 8-1.

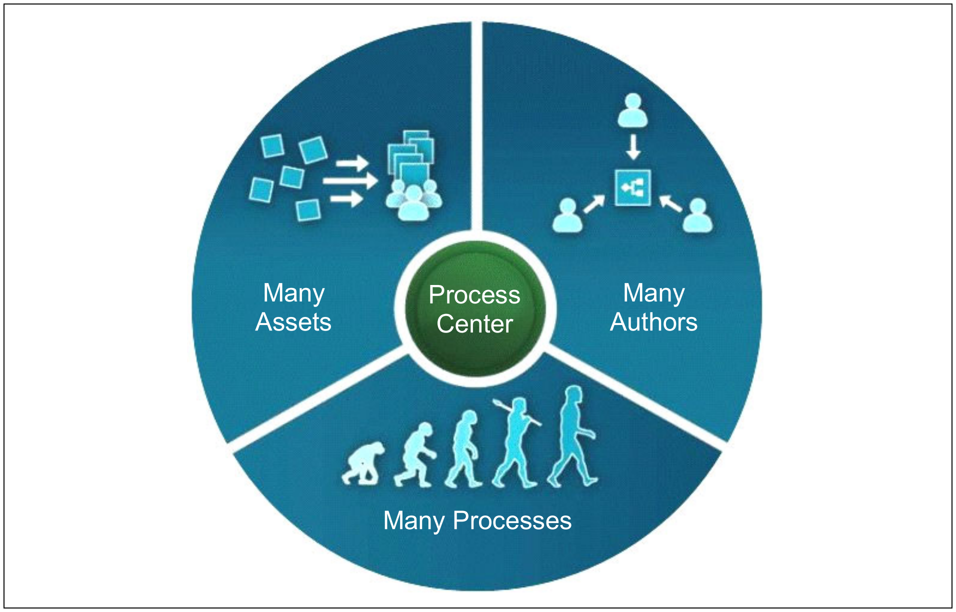


*Figure 8-1 Scaling up BPM*

Besides providing answers to these challenges, business process governance also aids in addressing the following situation:

- Promoting a new relationship between business and IT
- Consistency in measuring delivered business value
- Establishing consistent, data-driven decision-making processes
- Achieving regulatory compliance through process understanding, design, and consistent execution

The following sections introduce the concepts that comprise the suggested IBM business process governance model, including:

- Establishing strong *executive sponsorship*
- Establishing *BPM guiding principles* as a foundation for a *BPM operating model*
- Establishing a *BPM Center of Excellence*
- Establishing and filling the *BPM key roles in the BPM operating model*
- Establishing the *business process governance framework*

The first step in business process governance is to establish strong executive sponsorship.

## 8.2 Establishing strong executive sponsorship

Starting a BPM journey means introducing a new culture of change that requires vision and investment. Therefore, one key to a successful BPM program is to get buy-in at the highest executive level possible. Executive sponsorship should be obtained before the first BPM project is even started to ensure proper support and organizational engagement and to assist in managing the scope of the BPM journey.

The sponsorship of the top executive should be clear and publicized. IBM suggests that the executive declare the BPM project as a top priority for the department, division, or company as appropriate. The executive should be physically present to kick off the initial project, even if it is on the other side of the world. The engagement of the sponsoring executive establishes the priority of the transformational BPM projects among competing priorities and helps counter scope creep (“just one more requirement”) that is typical within iterative development efforts. Having the executive intervene and ask questions such as “What is critical to this release?” and “What can wait for the next iteration?” can be an effective way to reset project scope and expectations.

The first BPM project should deliver business value to the sponsoring executive and to the enterprise. In the top-down IBM approach, represented by the goal pyramid, the executive actively influences the project or program goals. The highest-level value metric for the project should be directly derived from these goals and be a metric for the sponsoring executive (for example, total cost and cycle time of the employee on-boarding process.) At the end of the project, the process owner reports back to the sponsoring executive on the project’s success and the value delivered to the enterprise.

## 8.3 Establishing BPM guiding principles

Guiding principles are statements for the BPM transformation that guide the execution of the projects, irrespective of which processes are currently being designed, developed, or deployed. They help process owners and process teams make decisions regarding moving forward with their solution design and delivery of the project.

We found the following examples of guiding principles to be a good starting point for any BPM program:

- BPM leadership, teams, skills, governance, and projects should be centralized to drive consistency of communication, education, and project execution in the first part of your BPM journey.
- BPM projects should be focused on delivering a greater customer experience and business value, not adding more steps or tools that increase process complexity.
- Process ownership is the key to BPM efforts. Strong process ownership is a requirement for overall success.
- The BPM projects strive to reduce the overall level of process and IT complexity (for example, integration is accomplished with SOA services instead of point-to-point integration whenever possible.)
- The BPM Center of Excellence (CoE) should be the team that acts on BPM projects and interlocks with the development (hopefully, service-oriented architecture) teams and infrastructure teams in IT to realize business value for the process owner and sponsoring executive.
- Business architecture, services architecture, and other guiding and regulating teams should provide governing guidance to the BPM CoE.
- The BPM CoE should develop guidance for when process owners and BPM solution teams need to engage control organizations (for example, compliance and risk) and other control-oriented organizations when redesigning processes, rules, or roles as part of a BPM solution.

It is expected that guiding principles are added or deleted from the list during an organization's BPM journey, as a reflection of the organization's culture maturing on its BPM journey.

## 8.4 Establishing the BPM operating model

This section describes the BPM operating model and how it can be applied to BPM projects.

## 8.4.1 The BPM operating model

The first step is to set up a BPM operating model (Figure 8-2) that supports business process governance throughout the BPM journey. The BPM CoE forms part of the operating model.

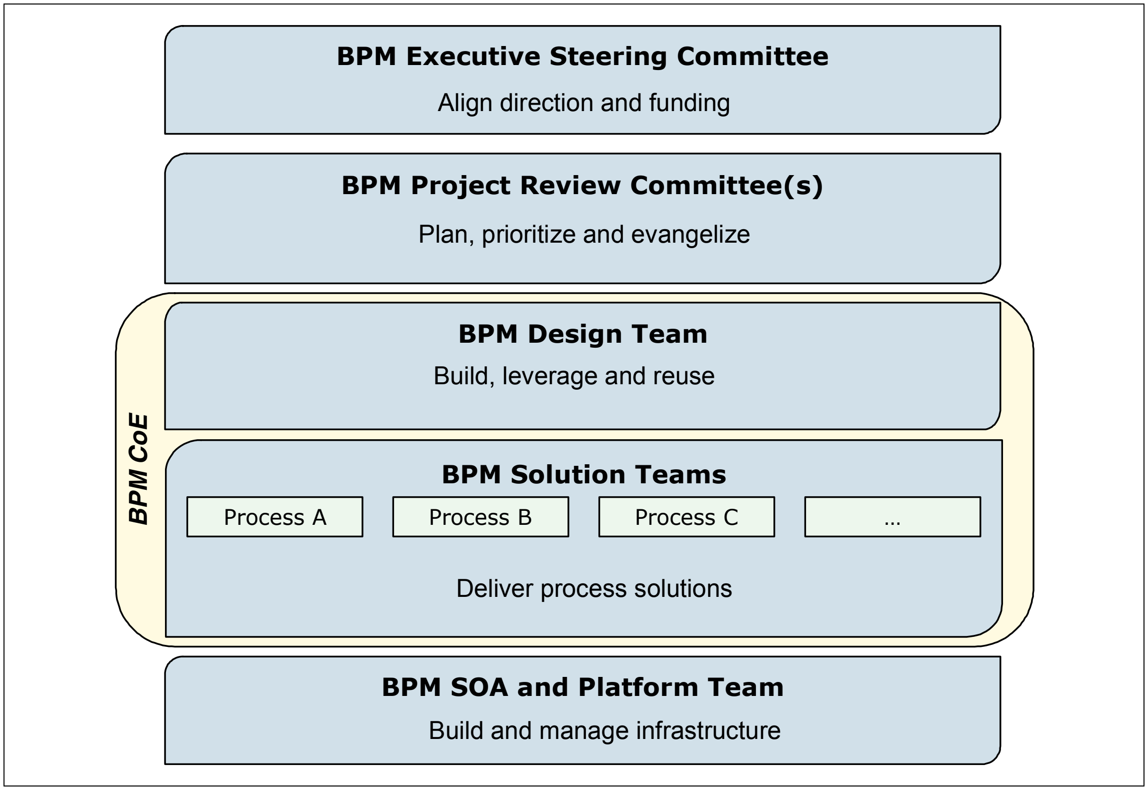

BPM Executive Steering Committee
Align direction and funding
BPM Project Review Committee(s)
Plan, prioritize and evangelize
BPM CoE
BPM Design Team
Build, leverage and reuse
BPM Solution Teams
Process A
Process B
Process C
...
Deliver process solutions
BPM SOA and Platform Team
Build and manage infrastructure

*Figure 8-2 BPM operating model*

### BPM executive steering committee

The BPM executive steering committee determines strategy, sets direction, and is responsible for the overall performance of the BPM program. This committee provides critical executive sponsorship, establishes the funding model, and allocates resources for all BPM projects. The funding model might not be established during the first engagement. The BPM executive steering committee also acts as a decision board for all issues related to strategy within ongoing projects. It is also in the BPM executive steering committee where process areas are proposed for consideration for future BPM solutions efforts.

The process owner (see “Process ownership”) participates in the BPM executive steering committee and in the BPM project review committees.

## BPM project review committee

The BPM project review committee meets to guide and advise in-progress BPM projects within a specific process area. More than one committee can be active at a time, because BPM projects can work on different process areas at the same time. The BPM project review committee acts as a decision-making board for current BPM projects where there is a risk for under-delivering on the original business value case.

The BPM project committee is responsible for the direction and allocation of CoE resources for BPM projects, driving consistent metrics across the BPM projects.

The BPM project review committee focuses on:

- Planning and prioritizing opportunities
- Establishing the enterprise process roadmap
- Evangelizing process ownership and BPM across the organization
- Standardizing methods
- Managing the skills development plan

## BPM design and architecture team

The charter of the BPM design team is to govern BPM projects to ensure that the tactical BPM solution aligns with the strategic vision of the organization’s business architecture. This solution includes developing measurement standards while meeting the business value case sought by the process owner.

The BPM design review teams are directed by the BPM CoE, but in composition they are simply virtual teams that assemble to ensure that architectural principles (business and technical) are adhered to across the full range of BPM projects. The lead business architect is responsible for assembling the appropriate team. The focus of the BPM design team is on consistency of delivery, strategic reuse, and inherent value creation for the enterprise.

The BPM design team is responsible for:

- Standardizing designs through enforcement of modeling standards and practices
- Identifying reuse potential for existing processes or subprocesses
- Overseeing the BPM design governance process and its execution
- Ensuring solution robustness without adding unnecessary complexity
- Mentorship and training of BPM practitioners along the way

### BPM solution teams

The charter of the BPM solution team is to deliver BPM projects in established time frames, to the process owner's satisfaction, and with (at a minimum) enough value creation to fulfill the value case. The focus is on consistency of modeling and development techniques, reduction in complexity, timely delivery, strategic reuse of technology, and overall value creation for the process owner and the enterprise. The BPM solution teams are directed by the BPM CoE.

The responsibilities of a BPM solution team include:

- Defining the process
- Scoping the iterative solution development effort
- Building and testing solutions
- Solution deployment and support
- Process solution risk management
- Process optimization
- Value case satisfaction

### BPM SOA team

The BPM SOA and platform team has two primary responsibilities:

- To help design, develop, and maintain the data services in support of the BPM solutions
- To plan, assemble, build, and manage the infrastructure in support of the BPM solutions

Managing the infrastructure includes monitoring its usage, configuring and tuning the infrastructure, and planning for future growth.

## 8.4.2 The operating model (governance) applied to a BPM project

Figure 8-3 on page 233 illustrates how the operating model is applied to a specific BPM project using the IBM iterative development approach. The steps throughout the lifecycle of the BPM project are:

1. The *business process governance lead* (likely, the lead business architect and member of the BPM CoE) schedules a business design review, following Playback 0.
2. The *process owner* presents the design and vision for the process, its metrics, and its value case to the BPM design and architecture team.
3. With approval from the governance lead, the process owner proceeds through Playback 1 and determines when the project is ready for the operational pilot.

4. During or at completion of Playback 2, the business process governance lead schedules a review of the services architecture.
5. The lead business architect and the process owner present their process designs and vision for the process and integrations, including metrics and value case, to the services architecture committee.
6. With the approval of the committee, and the process owner satisfied that enough development and integration is captured to make the business case, the process proceeds through Playback 3. The BPM solution team prepares to deploy the integration-enabled process to production (“go live.”)
7. After deployment to production, the effectiveness and value of the process is continuously monitored and reported up to the BPM CoE.

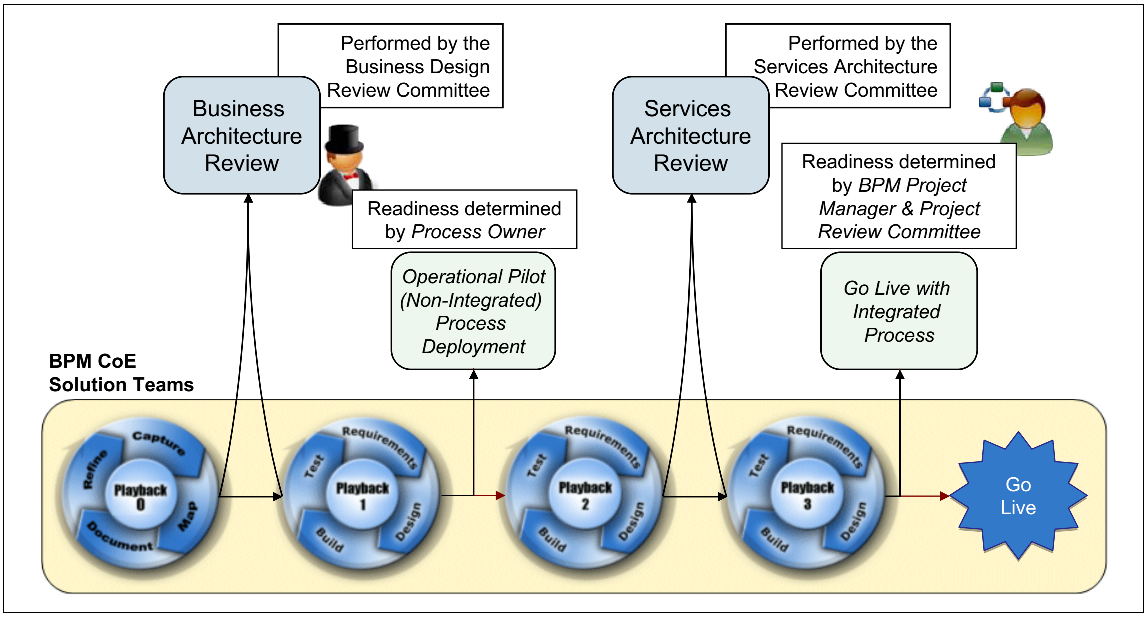

Performed by the Business Design Review Committee
Business Architecture Review
Readiness determined by *Process Owner*
*Operational Pilot (Non-Integrated) Process Deployment*
Performed by the Services Architecture Review Committee
Services Architecture Review
Readiness determined by *BPM Project Manager & Project Review Committee*
*Go Live with Integrated Process*
BPM CoE Solution Teams
Capture
Refine
Playback 0
Map
Document
Requirements
Test
Playback 1
Design
Build
Requirements
Test
Playback 2
Design
Build
Requirements
Test
Playback 3
Design
Build
Go Live

*Figure 8-3 A BPM project and the operational model*

## 8.5 Establishing the BPM Center of Excellence

The operating model explicitly includes a BPM CoE to direct the BPM design team and BPM solution teams, and to champion and evangelize BPM across the company. This section explores the BPM CoE in more detail and the challenges that it addresses and its goals and responsibilities.

### 8.5.1 Why we need a BPM CoE

When beginning your journey with BPM within an organization, you encounter challenges that your organization can address by establishing a BPM CoE. Typical challenges include:

- Projects under-delivering on the original business value case
- Alignment difficulties between business and IT
- Different methods and standards across BPM projects
- Lack of coordination within the organization, leading to suboptimal usage of resources

The previously mentioned points illustrate why strong executive sponsorship and effective working relationships between all departments in an organization are essential, and the charter of the BPM Center of Excellence is to ensure that these conditions exist.

### 8.5.2 The charter of the BPM Center of Excellence

The mission and goals of the BPM CoE are to ensure the success of the organization's BPM approach and initiatives by providing business solutions that support the organization's strategy and goals. The goals of the BPM CoE are:

- To provide a single entity for using standardized methods, tools, assets, skills, and resources for the organization
- To foster clear communication and coordination between business and IT along the BPM journey
- To ensure alignment with business architecture, process ownership, and the overall enterprise
- To provide and manage BPM policies and procedures
- To identify BPM roles and map them within the organization

### 8.5.3 Core responsibilities of the BPM CoE

The BPM CoE mission and goals provide a clear starting point in determining the core responsibilities:

- Managing and maintaining BPM portfolio planning, including coordination between BPM projects
- Managing the status of BPM projects

- Determining, implementing, and promoting the adoption of BPM standards and policies
- Tracking delivered BPM value through established metrics
- Managing and maintaining the deployed process models
- Supplying and maintaining skills for BPM initiatives
- Gathering and establishing best practices
- Maintaining and promoting reusable BPM assets
- Developing custom value-adding tools for the enterprise
- Fostering processing (and even business model) innovation
- Delivering BPM education

### 8.5.4 Evaluating the success of the BPM CoE

At the setup of the BPM CoE, the success criteria and the metrics to measure success should be included in the BPM CoE charter. This action ensures the focus and direction needed to maintain executive support. The BPM CoE metrics should be directly related to its responsibilities and characterize the value of the CoE.

The number of BPM projects for which the BPM CoE provides resources and consultancy are good candidates for determining return on investment (ROI) of the BPM CoE. The reuse of assets on BPM projects is in the same category. Less tangible, but valid, is the number of projects within budget, schedule, and with the expected business value delivered.

## 8.5.5 How the BPM CoE evolves

Typically, a BPM CoE is not established with the initial BPM project. It is established when the first BPM project is extended into a BPM program (Figure 8-4).

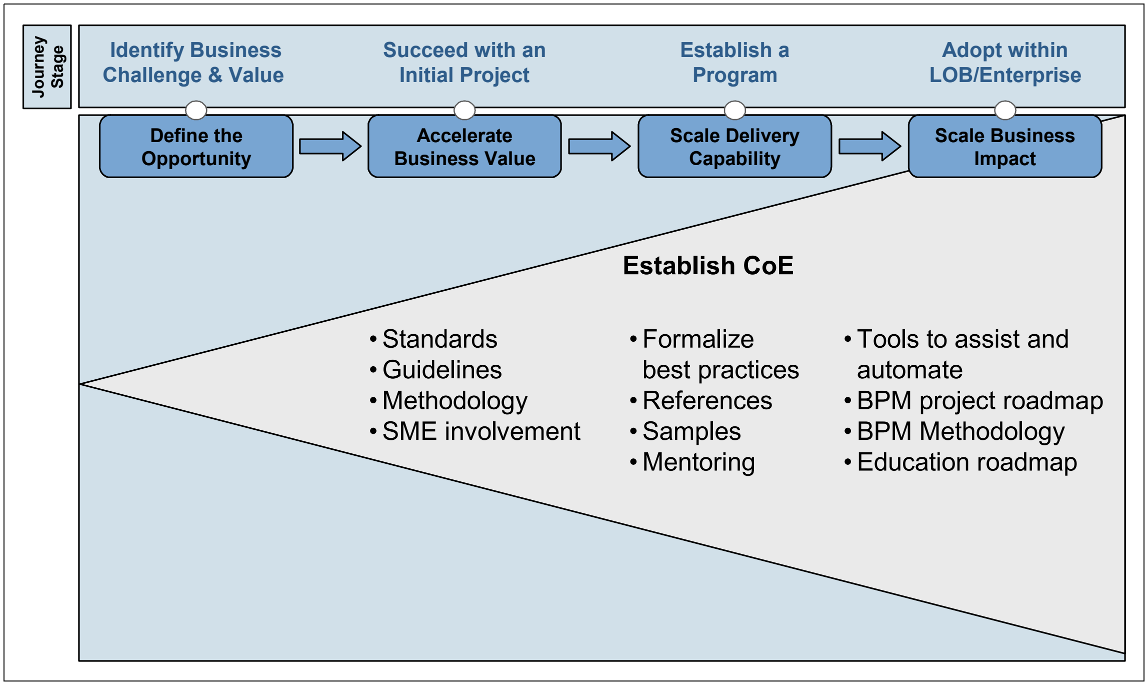


*Figure 8-4 Evolution of BPM CoE*

### Proposed approach for establishing the BPM CoE

The proposed approach for establishing a CoE includes the following phases:

1. Planning and establishing a roadmap
2. Implementing the roadmap
3. Activating the BPM CoE

#### *Planning and establishing a roadmap*

First, you need to socialize the BPM CoE and the business process governance and operational models. The obtained feedback is then incorporated into a formal charter for the BPM CoE. The BPM CoE charter contains:

- The mission
- The scope
- The roles and responsibilities
- The governance framework

- The mentoring and education plan
- The processes and procedures

The charter is then extended into a roadmap. The charter and roadmap are presented to the organization's BPM sponsor and stakeholders to obtain approval.

#### *Implementing the roadmap*

The implementation starts with identifying the CoE human resources and then training them on methods and tools. The operating model needs to be further formalized, and any questions about the working of the operating model need to be clarified.

Define the standard performance metrics that serve as a baseline for all BPM projects. Then you integrate the defined performance metrics into a common reporting process. Also include a change readiness assessment in the implementation of the roadmap.

#### *Activating the BPM CoE*

Start by deploying the CoE on the first BPM project. The feedback from this first BPM project is used to evaluate the contribution by CoE resources and to refine the operating model. You build upon the change readiness assessment from the implementation step to create and execute a change management plan.

## 8.6 BPM key roles

In this section, we describe the core participants in business process governance:

- BPM sponsor
- BPM process owner
- BPM CoE leader

### 8.6.1 BPM sponsor

The BPM sponsor role is to ensure support and engagement of all departments in the organization. The sponsor provides the strategy and goals of the BPM program. The sponsor sets the scope of the BPM journey. The sponsor ensures that all BPM projects are aligned with the overall goals and that they deliver business value. Because this role is critical to the success of the BPM journey, the BPM sponsor should be an executive person. The BPM sponsor in our Call Center Company C scenario is the chief executive officer (CEO.)

### 8.6.2 BPM process owner

The BPM process owner is the business representative who is assigned responsibility for a process value delivery across an organization, and who has the authority needed to fulfill that responsibility. A process owner has responsibility for the process from when it originates with the customer to where it ends with the customer and at all points in between. The process owner employs influence and gentle but firm management authority to ensure process coordination, execution, and improvement over time. The process owner thus has explicit interest in the success of the BPM project.

In summary, a process owner owns the end-to-end business process design, coordination between channels and individual departments, the process strategy and vision, and the process performance results. The process owner manages the process as an asset and maximizes the return on the company's investment, while advocating for customer value and the overall customer experience.

### 8.6.3 BPM CoE leader

The BPM CoE leader sets the direction for and oversees the CoE team and business SMEs. The CoE leader provides guidance in the areas of thought leadership, leading practices, process improvement method and tools, and overall strategic planning for achieving process discipline. The BPM CoE leader works closely with the BPM steering committee to develop decision criteria, process policies, issue resolution, and roles and responsibilities of the BPM steering committee. To fulfill the role, the BPM CoE leader needs subject matter expertise in process assessment and improvement methods and tools. The BPM CoE leader is a senior role that requires senior leadership experience, specifically in the area of transformation, and the ability to influence others.

## 8.7 Defining the business process governance framework

Business process governance is a catalyst for continuous process improvement and the collaboration between business and IT on corporate goals. Business process governance is essential to ensuring strong business and IT alignment by creating a framework that ensures the use of standard methods and tools to drive consistent and scalable delivery.

Business process governance contributes to *corporate governance* (Figure 8-5). Corporate governance is a framework for determining corporate direction and performance, how the corporation uses its resources, and what accountability exists for the stewardship of those resources.

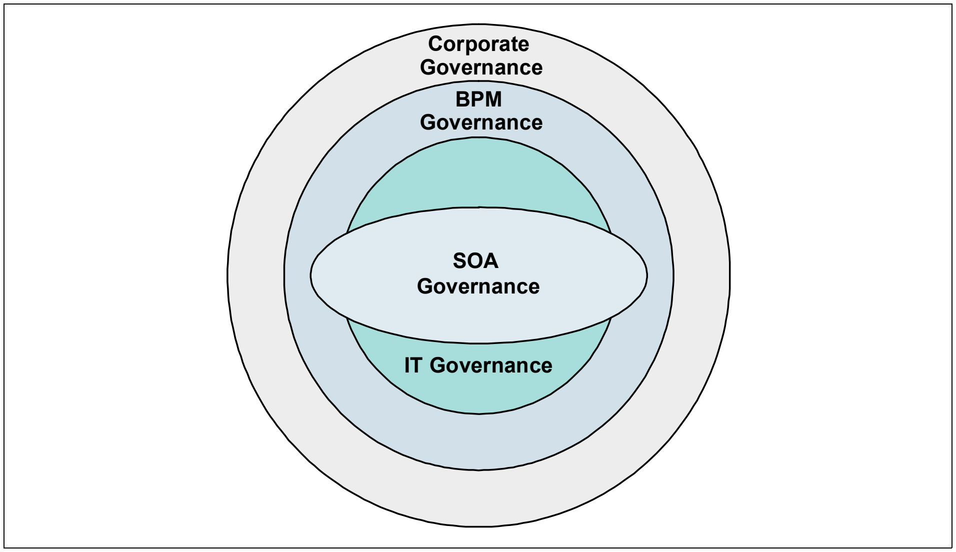


*Figure 8-5 Business process governance in the big picture*

*Business process governance* is a framework that ensures that the corporation's direction is realized at an operating level and is reflected through operating performance. The BPM layer interacts with IT, business, and SOA layers, and it requires a governance framework that accounts for those interactions. Given the complexity of these interactions, business process governance is an evolutionary process.

*IT governance* is a framework for establishing mechanisms and policies to ensure that the corporation's direction is supported and sustained by technology.

*SOA governance* is a specialization of IT governance that places key decisions within the context of the lifecycle of service components and services.

What constitutes a business process governance model? A business process governance model:

- Determines a joint vision
- Establishes policies, guidelines, standards, and frameworks
- Promotes enterprise-level performance metrics

- Specifies chains of responsibility, authority, and communication to empower people
- Establishes measurement, policy, and control mechanisms to empower people to carry out their roles and responsibilities
- Puts in place organizational structures for governing the lifecycle of processes
- Establishes procedures for using and sharing process knowledge to reduce process efforts
- Specifies BPM funding models
- Governs and measures the business value of processes
- Establishes a clear communication channel to enhance collaboration between business and IT

The following section describes in more detail certain key elements of the business process governance model.

### Process decomposition framework

The first critical step toward a *process decomposition framework* is building out a common taxonomy for BPM projects.

Multiple, parallel BPM efforts can create confusion and duplication of effort in the absence of a common business process taxonomy or vocabulary. A project or program requires awareness of the need to define the common taxonomy for efforts (for example, what defines a customer? Is it a client, a prospect, or any involved party?) and follows the conventions and rules accordingly. In mature industries, there are likely to be associations that already perform these functions (either as their mission or in a de-facto manner), so they are the best places to start, as they save time and reduce arguments within an organization. However, the beginning taxonomy should not be static. Like BPM projects, taxonomy and the meaning of certain terms should evolve as the organization’s BPM maturity evolves.

Establish a predefined *process hierarchy and decomposition*. Development time lines, and how well process requirements are fulfilled, are impacted by the level of detail to which processes are defined, captured, and for which BPM solutions are designed and developed (for example, level 2 versus level 6). It is essential to predefine a process hierarchy and the decomposition levels that are the framework and reference structure for the BPM projects and program.

A solid starting point for this hierarchy is APQC's various process decomposition frameworks built by industry. Many industry associations might provide good starting places as well. Also, like the common taxonomy, there is abundant room to evolve the process hierarchy and decomposition as the organization evolves in its BPM maturity.

Be careful not to equate an IT-focused capability map, component business model, or value net with a *process hierarchy*. Equating these items adversely impacts the BPM project in the following ways:

- It gives the participants the impression that the BPM effort is another IT project for new IT tools.
- It sets back the business or process architecture component of the BPM journey, as over several months project participants realize that after processes are defined, you need to string together many elements of a value net to deliver any process. Processes do not equal capabilities, and they need to be organized appropriately to deliver enterprise value.

### *Next critical step*

Deploy a rigorous model management approach. The process-centric approach to BPM projects requires model management discipline for the maintenance, versioning, and collaboration of process models across the enterprise. It is best to realize and plan for this contingency at the beginning of the BPM journey, and anticipate the need going forward. These practices tie in tightly with the process hierarchy and decomposition, as it becomes the index for organizing the models that need the maintenance, versioning, and collaboration management guidelines this step delivers.

## Process ownership

Process ownership is a crucial factor in successful BPM solution deployment, adoption, and sustainability. A process-based ownership model is process-oriented (meaning that it is focused on all steps from a customer's request to fulfilling that request) and this model works cross line of business (LOB). A process-based ownership model:

- Solves customer and enterprise pain points across business processes and aligns with strategic goals.
- Manages the process through process ownership within an organization and horizontal alignment.
- Creates accountability and empowerment to enable process performance and results.
- Creates reusable enterprise-level processes.

- *White space* is minimized as functions are aligned to support specific processes.
- Over time, the business becomes governed and measured by the process.

### *Why process ownership is so important*

Through the process-based ownership model, you achieve visibility into the process contribution to the enterprise strategy, and you achieve the goals at an executive level. This executive visibility ensures that the end-to-end processes perform well. Clearly established ownership allows resolution of conflicting goals among processes through governed collaboration across departments.

### *What the potential pitfalls are*

The common pitfalls are those actions and compromises that organizations often make that reduce the authority and accountability of the process owner. Strong process ownership is crucial. The process owner can never be a temporary one. For the same reason, assigning a technical leader or a junior leader to the role instead of business representative is inviting additional risk into your BPM journey.

## Transformation cadence and the strategic plan

IBM advocates a rapid prototyping (agile) methodology for BPM development efforts. In doing so, how an organization attacks its projects and programs often becomes more important than what it attacks, because the overall goal is to focus on value. In doing so, the technique of time boxing the efforts to focus on value delivered over short time bursts is critical. This situation also builds up a *transformation cadence* that many organizations lack when they attempt to transform their processes, their development techniques, their delivery models, and even their business models. This cadence, tied into the strategic plan and roadmap, and executive sponsorship and the focus on value (not technology solutions), is critical to the BPM transformation journey, the success of the projects, and the return on the BPM investment.

## Process KPIs, metrics, and measurement

The BPM Center of Excellence establishes standard performance metrics as a baseline for all BPM-approved projects to ensure that projects are tracked throughout their lifecycle against the proposed business case. The metrics are integrated into a common reporting process. Continuous monitoring of the projects throughout their lifecycle is important. It ensures that the BPM projects deliver the expected business value.

# Abbreviations and acronyms

| | |
|---|---|
| **BAM** | Business Activity Monitoring |
| **BLA** | Business Level Application |
| **BPD** | Business Process Definition |
| **BPM** | Business Process Management |
| **BPMS** | Business Process Management System |
| **BSOR** | Business Data System of Record |
| **COE** | Center Of Excellence |
| **CPI** | Continuous Process Improvement |
| **ECM** | Enterprise Content Management |
| **EPV** | Exposed Process Variable |
| **ESB** | Enterprise Service Bus |
| **ETL** | Extract Transform and Load |
| **HR** | Human Resources |
| **IBM** | International Business Machines Corporation |
| **ITSO** | International Technical Support Organization |
| **KPI** | Key Performance Indicator |
| **LOB** | Line Of Business |
| **ROI** | Return On Investment |
| **SCA** | Service Component Architecture |
| **SLA** | Service Level Agreement |
| **SME** | Subject Matter Expert |

# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

## IBM Redbooks

For information about ordering these publications, see “Help from IBM” on page 246. Note that some of the documents referenced here may be available in softcopy only.

- *Human-Centric Business Process Management with WebSphere Process Server V6*, SG24-7477

## Other publications

These publications are also relevant as further information sources:

McConnel, *Software Estimation: Demystifying the Black Art*, Microsoft Press, 2006, ISBN 0735605351

McConnel, *Software Project Survival Guide*, Micosoft Press, 1997, ISBN 1572316217

## Online resources

The following websites are also relevant as further information sources:

- Business process modeling

  http://bpmwiki.blueworkslive.com/display/commwiki/Five+Guidelines+to+Better+Process+Modeling
- Cookbook for deploying business processes

  http://bpmwiki.blueworkslive.com/display/commwiki/Cookbook+for+deploying+a+business+process

- General process design best practices

  http://bpmwiki.blueworkslive.com/display/commwiki/Best+Practices+Recommendations
- IBM Blueworks Live

  https://blueworkslive.com/
- IBM Business Process Manager design patterns

  http://bpmwiki.blueworkslive.com/display/commwiki/Design+Patterns
- Integration technologies

  http://bpmwiki.blueworkslive.com/display/commwiki/Integration+Technologies
- Testing

  http://bpmwiki.blueworkslive.com/display/commwiki/Testing

# Help from IBM

IBM Support and downloads

**ibm.com**/support

IBM Global Services

**ibm.com**/services

IBM Redbooks

Scaling BPM Adoption: From Project to Program with IBM Business Process Manager

# Scaling BPM Adoption

## From Project to Program with IBM Business Process Manager

**Embrace process awareness as your roadmap for change**

**Design robust business processes that scale with your needs**

**Succeed with prescriptive methods and guidelines**

Your first Business Process Management (BPM) project is a crucial first step on your BPM journey. It is important to begin this journey with a philosophy of change that allows you to avoid common pitfalls that lead to failed BPM projects, and ultimately, poor BPM adoption. This IBM Redbooks publication describes the methodology and best practices that lead to a successful project and how to use that success to scale to enterprise-wide BPM adoption. This updated edition contains a new chapter on planning a BPM project.

The intended audience for this book includes all people who participate in the discovery, planning, delivery, deployment, and continuous improvement activities for a business process. These roles include process owners, process participants, subject matter experts (SMEs) from the operational business, and technologists responsible for delivery, including BPM analysts, BPM solution architects, BPM administrators, and BPM developers.

SG24-7973-01 ISBN 073843681X